FORMAL DYNAMICAL SYSTEMS AND NUMERICAL ALGORITHMS①

形式动力系统与数值算法

§1 Preliminaries

We study vector fields, their associated dynamical systems, phase flows together with their algorithmic approximations in $\mathbf{R}^N$ from the formal power series approach. Our considerations will be local in both space and time, all related objects are $\mathbf{C}^\infty$ smooth. We use coordinate description and matrix notation, the coordinate vectors in $\mathbf{R}^N$ and vector functions $a : \mathbf{R}^N \to \mathbf{R}^N$ are denoted by column matrices. The identity vector function 1_N is given by $1_N(x) = x$. For vector function $a = (a_1, \cdots, a_N)^T : \mathbf{R}^N \to \mathbf{R}^N$,

$$\begin{aligned} a_* &:= \left(\frac{\partial a_i}{\partial x_j}\right) = \text{ Jacobian matrix of } a \\ a^* &:= \sum a_i \frac{\partial}{\partial x_i} = \text{ linear differential operator of first order associated to } a. \end{aligned}$$

The association $a \to a^*$ is linear, a^* operates on scalar functions $\phi : \mathbf{R}^N \to \mathbf{R}$ and on vector functions $b : \mathbf{R}^N \to \mathbf{R}^N$ as

$$\begin{aligned} a^*\phi &= \sum a_i \frac{\partial \phi}{\partial x_i} \\ a^*b &= a^*(b_1, \cdots, b_N)^T = (a^*b_1, \cdots, a^*b_N)^T = b_*a, \quad a^*1_N = a. \end{aligned}$$

Multiple application of linear differential operators are naturally defined such as a^*b^*, $(a^*b^*)c^*$, $a^*(b^*c^*)$, etc. The operations are multilinear, associative but non-commutative; thus powers can be defined

$$a^{*k} = a^*a^* \cdots a^*(k \text{ times}), \qquad a^k := a^{*k}1_N$$

The identity operator I operates on scalar and vector functions ϕ and b as $I\phi = \phi$, $Ib = b$.

We identify all vector functions $a : \mathbf{R}^N \to \mathbf{R}^N$ as vector fields. All vector fields in $\mathbf{R}^N$ form a (∞-dimensional) real Lie algebra $\mathbf{V}^N$ under Lie bracket

$$[a, b] := a_*b - b_*a = b^*a - a^*b = (b^*a^* - a^*b^*)1_N.$$

The Lie algebra $\mathbf{V}^N$ is associated to the (∞-dimensional) local Lie group $\mathbf{D}^N$ of near-identity diffeomorphisms—or simply near-1 maps—of $\mathbf{R}^N$.

---

①In Feng K, Shi Z C, ed. Proc Conf on Computation of Differential Equations and Dynamical Systems. Singapore: World Scientific, 1993. 1–10

Consider the dynamical system in $\mathbf{R}^N$

$$\frac{dx}{dt} = a(x) \tag{1}$$

defined by a vector field a. It possesses a phase flow $e_a^t = e^t$, which is a one-parameter (in t) group of near-1 maps of $\mathbf{R}^N$,

$$e^0 = 1_N, \qquad e^{t+s} = e^t \circ e^s,$$

and generates the solution by $x(0) \to e_a^t x(0) = x(t)$. The phase flow is expressible as a convergent power series in t,

$$e_a^t = 1_N + \sum_{k=1}^{\infty} t^k e_k,$$

$$e_0 = 1_N, \qquad e_k = \frac{1}{k} a^* e_{k-1} = \frac{1}{k!}(a^*)^k 1_N = \frac{1}{k!} a^k.$$

We define

$$\operatorname{Exp} ta^* := I + \sum_{k=1}^{\infty} \frac{1}{k!}(ta)^{*k}, \quad I \text{ is the identity operator,} \tag{2}$$

This is an operator power series operating on scalar functions and vector functions, and define

$$\exp ta := (\operatorname{Exp} ta^*)1_N = 1_N + \sum_{k=1}^{\infty} \frac{1}{k!}(ta^*)^k 1_N = 1_N + \sum_{k=1}^{\infty} \frac{t^k}{k!} a^k. \tag{2)'$$

then

$$\begin{cases} e_a^t = (\operatorname{Exp} ta^*)1_N = \exp ta, \\ \phi \circ e_a^t = \phi \circ \exp ta = (\operatorname{Exp} ta^*)\phi, \quad \forall \text{ scaler function } \phi, \\ b \circ e_a^t = b \circ \exp ta = (\operatorname{Exp} ta^*)b = (\operatorname{Exp} ta^*)b^* 1_N, \quad \forall \text{ vector function } b. \end{cases} \tag{3}$$

Each numerical algorithm solving the system (1) possesses the ***step transition map*** f_a^s which is one-parameter (in step-size s) family (in general not a one-parameter group in s) of near-1 maps on R^N, expressible as a convergent power series in s

$$f_a^s = 1_N + \sum_{1}^{\infty} s^k f_k, \tag{4}$$

the coefficients can be determined recursively form the defining difference equation. The transition generates the numerical solution $x(0) \to (f_a^s)^N x(0) \approx x(Ns)$ by iterations with step-size s chosen fixed in general.

The main problem is to construct and analyse the algorithmic approximations $f_a^s \approx e_a^t\big|_{t=s} = e_a^s$ in a ***proper*** way. For this purpose we propose a ***unified*** framework based on the apparatus of formal power series, Lie algebra of vector fields and the corresponding Lie group of diffeomorphisms [1,2].

§2 Near-0 and Near-1 Formal Power Series

Among the formal power series $\sum_0^\infty s^k a_k$, $a_k : \mathbf{R}^N \to \mathbf{R}^N$, we pick out two special classes. The first class consists of those with $a_0 = 0$, called near-0 formal vector fields; the second class consists of those with $a_0 = 1_N$, called near-1 formal maps (diffeomorphisms).

All near-0 formal vector fields $a^s = \sum_1^\infty s^k a_k$ form a (∞-dim.) real Lie algebra $\mathbf{FV}^N$ under the Lie bracket

$$[a^s, b^s] = \Big[\sum_1^\infty s^k a_k, \sum_1^\infty s^k b_k\Big] := \sum_{k=2}^\infty s^k \sum_{i+j=k} [a_i, b_j].$$

The associated near-0 formal differential operators and their products are

$$\begin{aligned} (a^s)_* &:= \Big(\sum_1^\infty s^k a_k\Big)_* := \sum_1^\infty s^k a_{k*} \\ a^{s*} &= \Big(\sum_1^\infty s^k a_k\Big)^* := \sum_1^\infty s^k a_k^*, \\ a^{s*} b^{s*} &:= \sum_2^\infty s^k \sum_{i+j=k} a_i^* b_j^*, \quad (a^{s*})^2 := a^{s*} a^{s*}, \text{ etc.} \end{aligned}$$

For any vector function $a = (a_1, \cdots, a_N)^T : \mathbf{R}^N \to \mathbf{R}^N$ and any near-1 formal map $g^s = 1 + \sum s^k g_k$ we define the composition

$$\begin{aligned} (a \circ g^s)(x) &= a(g^s(x)) = a(x) + \sum_1^\infty s^k (a \circ g)_k(x), \\ (a \circ g)_k &= \sum_{m=1}^k \sum_{k_1 + \cdots + k_m = k} \frac{1}{m!} (D^m a)(g_{k_1}, \cdots, g_{k_m}), \end{aligned} \tag{5}$$

where

$$\begin{aligned} D^m a &= (D^m a_1, \cdots, D^m a_N)^T, \\ D^m a_i(v_1, \cdots, v_m) &= \sum_{j_1, \cdots, j_m = 1}^N \frac{\partial^m a_i}{\partial x_{j_1} \cdots \partial x_{j_m}} v_{1 j_1} \cdots v_{m j_m} \end{aligned}$$

is the usual m-th differential multi-linear form for m tangent vectors $v_i = (v_{i_1}, \cdots, v_{i_N})^T$, $i = 1, \cdots, m$ at point $x \in \mathbf{R}^N$ which is invariant under permutation of vectors. Using the identities

$$\begin{aligned} (D^1 a)(b) &= b^* a, \\ (D^2 a)(b, c) &= (c^* b^* - (c^* b)^*) a, \\ (D^3 a)(b, b, b) &= (b^{*3} + 2b^{3*} - 3b^* b^{2*}) a, \end{aligned}$$

we get in particular

$$\begin{aligned} (a \circ g)_1 &= g_1^* a, \\ (a \circ g)_2 &= g_2^* a + \frac{1}{2}(g_1^{*2} - g_1^{2*}) a, \\ (a \circ g)_3 &= g_3^* a + ((g_2^* g_1^* - (g_2^* g_1)^*) a + \frac{1}{3!}(g_1^{*3} + 2g_1^{3*} - 3g_1^* g_1^{2*}) a. \end{aligned}$$

For any two near-1 formal maps $f^s = 1_N + \sum s^k a_k$, $g^s = 1 + \sum s^k g_k$, the composition $f^s \circ g^s$ is defined in a term by term way:

$$
\begin{aligned}
&(f^s \circ g^s)(x) = f^s(g^s(x)) = 1_N(g^s(x)) + \sum_{k=1}^{\infty} s^k f_k(g^s(x)) \\
&\qquad\qquad =: 1_N(x) + \sum_{k=1}^{\infty} s^k (f \circ g)_k(x), \\
&(f \circ g)_1 = f_1 + g_1, \\
&(f \circ g)_k = f_k + g_k + \delta(f_1, \cdots, f_{k-1}; g_1, \cdots, g_k), \quad k \geq 2, \\
&\delta(f_1, \cdots, f_{k-1}; g_1, \cdots, g_{k-1}) = \sum_{i=1}^{k-1} \sum_{m=1}^{i} \sum_{i_1 + \cdots + i_m = i} \frac{1}{m!} (D^m f_{k-i})(g_{i_1}, \cdots, g_{i_m}),
\end{aligned}
\tag{6}
$$

we get in particular,

$$
\begin{aligned}
(f \circ g)_2 &= f_2 + g_2 + g_1^* f_1, \\
(f \circ g)_3 &= f_3 + g_3 + g_1^* f_2 + g_2^* f_1 + \frac{1}{2}(g_1^{*2} - g_1^{2*}) f_1, \\
(f \circ g)_4 &= f_4 + g_4 + g_1^* f_3 + g_2^* f_2 + g_3^* f_1 + \frac{1}{2}(g_1^{*2} - g_1^{2*}) f_2 \\
&\quad + (g_2^* g_1^* - (g_2^* g_1)^*) f_1 + \frac{1}{3!}(g_1^{*3} + 2 g_1^{3*} - 3 g_1^* g_1^{2*}) f_1.
\end{aligned}
$$

Under this composition rule, all near-1 formal maps $f^s = 1 + \sum_1^{\infty} s^k f_k$ form (∞-dim) formal Lie group $\mathbf{FD}^N$. In group $\mathbf{FD}^N$ inverse elements, square roots, rational powers, etc., always exist, their coefficients can always be determined recursively by the defining composition relations. For example, the inverse $(f^s)^{-1} := 1 + \sum s^k h_k = h^s$ is defined by $(f^s \circ h^s) = 1_N$, hence

$$
f_1 + h_1 = 0, \quad f_k + h_k + \delta(f_1, \cdots, f_{k-1}; h_1, \cdots, h_{k-1}) = 0, \quad k \geq 2.
$$

In particular,

$$
h_1 = -f_1, \quad h_2 = -f_2 + f_1^2, \quad h_3 = -f_3 + f_1^* f_2 + (f_2^* - f_1^{2*}) f_1 - \frac{1}{2} f_1^3 + \frac{1}{2} f_1^{2*} f_1.
$$

There is an obvious one-one correspondence between the Lie algebra $\mathbf{FV}^N$ and the Lie group $\mathbf{FD}^N$, established simply by $+1_N$ and by -1_N. However, the more significant one-one correspondence between them is given by exp and its inverse log.

$$
\begin{aligned}
&\exp : \mathbf{FV}^N \to \mathbf{FD}^N \\
&a^s = \sum_1^{\infty} s^k a_k \to \exp a^s := 1_N + \sum_{m=1}^{\infty} \frac{1}{m!} (a^{s*})^m 1_N =: 1_N + \sum_1^{\infty} s^k f_k = f^s.
\end{aligned}
\tag{7}
$$

Note that

$$
(a^{s*})^m = \left(\sum s^{k_1} a_{k_1}^*\right) \cdots \left(\sum s^{k_m} a_{k_m}^*\right) = \sum_{k_1, \cdots, k_m = 1}^{\infty} s^{k_1 + \cdots + k_m} a_{k_1}^* \cdots a_{k_m}^*,
$$

so we get easily

$$
\begin{aligned}
f_k &= \sum_{m=1}^{k} \frac{1}{m!} \sum_{k_1+\cdots+k_m=k} a^*_{k_1} \cdots a^*_{k_m} 1_N, \qquad k \geq 1, \quad f_1 = a_1, \qquad (8)\\
f_k &= a_k + \sum_{m=2}^{k} \frac{1}{m!} \sum_{k_1+\cdots+k_m=k} a^*_{k_1} \cdots a^*_{k_m} 1_N, \qquad k \geq 2, \quad f_2 = a_2 + \frac{1}{2} a_1^2.
\end{aligned}
$$

Note that (8) provides a 2-way recursion formula from $a_1, \cdots, a_k$ to $f_1, \cdots, f_k$ and vice versa. Therefore exp maps $\mathbf{FV}^N$ one-one onto $\mathbf{FD}^N$ and its inverse, i.e., log is defined by the same (8):

$$
\log = (\exp)^{-1} : \mathbf{FD}^N \to \mathbf{FV}^N, \quad \log \exp a^s = a^s, \quad \exp \log f^s = f^s.
$$

In particular

$$
\begin{aligned}
a_1 &= f_1, \quad a_2 = f_2 - \frac{1}{2} a_1^2, \quad a_3 = f_3 - \frac{1}{2}(a_1^* a_2 + a_2^* a_1) - \frac{1}{3!} a_1^3,\\
a_4 &= f_4 - \frac{1}{2}(a_1^* a_3 + a_2^2 + a_3^* a_1) - \frac{1}{3!}(a_1^* a_1^* a_2 + a_1^* a_2^* a_1 + a_2^* a_1^* a_1) - \frac{1}{4!} a_1^4, \qquad (9)\\
a_k &= f_k - \sum_{m=2}^{k-1} \frac{1}{m!} \sum_{k_1+\cdots+k_m=k} a^*_{k_1} \cdots a^*_{k_m} 1_N - \frac{1}{k!} a_1^k, \quad k \geq 3.
\end{aligned}
$$

An equivalent construction of $\log f^s = a^s$ is

$$
\log f^s = \sum_{m=1}^{\infty} \frac{(-1)^{m-1}}{m} h^s_m, \quad \text{where } h^s_1 = f^s - 1_N, \quad h^s_m = h^s_{m-1} \circ f^s - h^s_{m-1}. \qquad (10)
$$

It is easy to compute

$$
\begin{aligned}
h^s_1 &= \sum_{k=1}^{\infty} s^k f_k = \sum_{k_1=1}^{\infty} s^{k_1} (1_N \circ f)_{k_1}\\
h^s_2 &= \sum_{k_1=k_2=1}^{\infty} s^{k_1+k_2} ((1_N \circ f)_{k_1} \circ f)_{k_2},\\
h^s_3 &= \sum_{k_1,k_2,k_3=1}^{\infty} s^{k_1+k_2+k_3} (((1_N \circ f)_{k_1} \circ f)_{k_2} \circ f)_{k_3}, \text{ etc.}
\end{aligned}
$$

Substituting in (10) and equating with $\sum_1^{\infty} s^k a_k$, we get

$$
a_k = \sum_{m=1}^{k} \frac{(-1)^{m-1}}{m} \sum_{k_1+\cdots+k_m=k} (\cdots((1_N \circ f)_{k_1} \circ f)_{k_2} \cdots \circ f)_{k_m}. \qquad (11)
$$

It is easy to verify $\log \exp a^s = a^s$ for this log, so this is precisely the inverse of exp, thus agrees with the previous one.

We use the above construction (11) to establish the *formal* Campbell-Haus-dorff formula:
For arbitrary near-1 formal maps f^s, g^s

$$
\log(f^s \circ g^s) = \log f^s + \log g^s + \sum_{k=1}^{\infty} d_k(\log f^s, \log g^s), \qquad (12)
$$

where, for $\log f^s = a^s$, $\log g^s = b^s$,

$$d_k(a^s, b^s) = \frac{1}{k}\sum_{m=1}^{k}\frac{(-1)^{m-1}}{m}\sum_{\substack{p_1+q_1+\cdots+p_m+q_m=k\\ p_i+q_i\geq 1,\, p_i\geq 0,\, q_i\geq 0}}\frac{[(a^s)^{p_1}(b^s)^{q_1}\cdots(a^s)^{p_m}(b^s)^{q_m}]}{p_1!q_1!\cdots p_m!q_m!} \tag{13}$$

where

$$(x)^p = xx\cdots x \ (p\text{-times}), \quad [x_1x_2x_3\cdots x_n] = [[\cdots[[x_1, x_2], x_3], \cdots], x_n].$$

In particular

$$d_1 = \frac{1}{2}[a^s, b^s], \quad d_2 = \frac{1}{12}([a^sb^sb^s] + [b^sa^sa^s]), \quad d_3 = -\frac{1}{24}[a^sb^sb^sa^s].$$

Let $\log(f^s \circ g^s) = c^s = \sum_1^\infty s^k c_k$. Then

$$\begin{aligned}
c_1 &= a_1 + b_1, \qquad c_2 = a_2 + b_2 + \frac{1}{2}[a_1b_1],\\
c_3 &= a_3 + b_3 + \frac{1}{2}([a_1b_2] + [a_2b_1]) + \frac{1}{12}([a_1b_1b_1] + [b_1a_1a_1]),\\
c_4 &= a_4 + b_4 + \frac{1}{12}([a_1b_3] + [a_2b_2] + [a_3b_1])\\
&\quad + \frac{1}{12}([a_1b_1b_2] + [a_1b_2b_1] + [a_2b_1b_1] + [b_1a_1a_2] + [b_1a_2a_1] + [b_2a_1a_1])\\
&\quad - \frac{1}{24}[a_1b_1b_1a_1], \quad \text{etc.}
\end{aligned}$$

Note that the classical CH formula is restricted to the composition of two one-parameter groups, where $\log f^s = sa_1$, $\log g^s = sb_1$.

The log transform reduces matters at the Lie group level to those at the easier level of Lie algebra. All properties of near-1 formal maps have their logarithmic interpretations. We list some of them, let $\log f^s = a^s = \sum_1^\infty s^k a_k$:

1. f^s is a phase flow, i.e., $f^{s+t} = f^s \circ f^t \iff \log f^s = sa_1$.
2. f^s is revertible, i.e., $f^s \circ f^{-s} = 1_N \iff \log f^s$ is odd in s.
3. f^s raised to real μth power $(f^s)^\mu \iff \log(f^s)^\mu = \mu \log f^s$. In particular $\log(f^s)^{-1} = -\log f^s$, $\log\sqrt{f^s} = \frac{1}{2}\log f^s$.
4. f^s scaled to $f^{\alpha s} \iff \log(f^{\alpha s}) = (\log f)^{\alpha s}$. In particular $\log(f^{-s}) = (\log f)^{-s}$.
5. $f^s - g^s = O(s^{p+1}) \iff \log f^s - \log g^s = O(s^{p+1})$.
6. $f^s \circ g^s = g^s \circ f^s \iff [\log f^s, \log g^s] = 0 \iff \log(f^s \circ g^s) = \log f^s + \log g^s$.
7. $(f^s \circ g^s) = h^s \iff \log h^s = \log(f^s \circ g^s) = \log f^s + \log g^s + \sum_1^\infty d_k(\log f^s, \log g^s)$.
8. f^s symplectic $\iff$ all a_k are Hamiltonian fields.
9. f^s contact $\iff$ all a_k are contact fields.
10. f^s volume-preserving $\iff$ all a_k are source-free fields.

The log transform has important bearing on dynamical systems with Lie algebra structure, the structure-preserving property of maps f^s at the Lie group ($\mathbf{G} \subset \mathbf{D}_m$) level can be characterized through their logarithms at the associated Lie algebra ($\mathbf{L} \subset \mathbf{V}_m$) level, see [2, 6].

§3 Algorithmic Approximations to Phase Flows

We return to the main problem of approximation to the phase flow for dynamical system $dx/dt = a(x)$.

$$f_a^s = f^s = 1_N + \sum s^k f_k \approx e_a^s = 1_N + \sum s^k e_k, \quad e_k = a^k/k!. \tag{14}$$

If $f_k = e_k$, $1 \leq k \leq p$, we say f_a^s is accurate to order $\geq p$, if moreover, $f_{p+1} \neq e_{p+1}$, we say it is accurate to order p.

Let $\log f^s = a^s = \sum s^k a_k$. Note that the first $p+1$ equations in (8) completely determine $a_1, a_2, \cdots, a_{p+1}$ and $f_1, f_2, \cdots, f_{p+1}$ each other. It is then easy to establish

$$\begin{aligned} &f_k = e_k, \quad 1 \leq k \leq p; \quad f_{p+1} \neq e_{p+1} \Longleftrightarrow \\ &a = a_1 = e_1; \quad a_k = 0, \quad 1 < k \leq p; \quad a_{p+1} = f_{p+1} - e_{p+1} \neq 0. \end{aligned} \tag{15}$$

So the orders of approximation for $f_a^s \approx e_a^s$ and for $\log f_a^s - sa$ are the same.

Moreover, note that we have a *formal field*

$$s^{-1} \log f^s = s^{-1} a^s = a + \sum_1^\infty s^k a_{k+1} = a + O(s^p) \tag{16}$$

which is the original field a up to a near-0 perturbation and defines a *formal dynamical system*

$$\frac{dx}{dt} = (s^{-1} \log f^s)(x) = a(x) + \sum_1^\infty s^k a_{k+1}(x) \tag{17}$$

having a *formal phase flow* (in two parameters t and s with group property in t) $e^t_{s^{-1}a^s} = \exp ts^{-1}a^s$ whose *diagonal formal flow* $e^t_{s^{-1}a^s}|_{t=s}$ is exactly f^s. This means that any compatible algorithm f_a^s of order p gives perturbed solution of a right equation with field a; however, it gives the right solution of a perturbed equation with field $s^{-1}\log f_a^s = a + O(s^p)$. There could be many methods with the same formal order of accuracy but with quite different qualitative behavior. The problem is to choose among them those leading to allowable perturbations in the equation. For systems with geometric structure, the propositions 8, 9 and 10 provide guidelines for a proper choice. The structure-preservation requirement for the algorithms precludes all unallowable perturbations alien to the pertinent type of dynamics. Take, for example, Hamiltonian systems. A transition map f_a^s for Hamiltonian field a is symplectic if and only if all fields a_k are Hamiltonian, i.e., the induced perturbations in the equation are Hamiltonian [2,6]. So symplectic algorithms are clean, inherently free from all kinds of perturbations alien to Hamiltonian dynamics (such as artificial dissipation inherent in the vast majority of conventional methods), this accounts for their superior performance. The situations are the same for contact and volume-preserving algorithms [3–6].

Finally we give, as an illustration, four simplest methods together with step transition maps and their logarithms.

$$e_a^s = 1_N + sa + \frac{1}{2}s^2a^2 + \frac{1}{3!}s^3a^3 + O(s^4).$$

1. Explicit Euler method E

$$\begin{aligned}
&x_1 - x_0 = sa(x_0),\\
&f^s - 1_N = sa,\\
&f_E^s = 1_N + sa,\\
&\log f_E^s = sa - \frac{s^2}{2}a^2 + O(s^3),
\end{aligned}$$

non-revertible, ord $= 1$.

2. Implicit Euler method I

$$\begin{aligned}
&x_1 - x_0 = sa(x_1),\\
&f^s - 1_N = sa \circ f^s,\\
&f_I^s = (1_N - sa)^{-1} = (f_E^{-s})^{-1} = sa + s^2a^2 + O(s^3),\\
&\log f_I^s = sa + \frac{s^2}{2}a^2 + O(s^3),
\end{aligned}$$

non-revertible, ord $= 1$.

3. Trapezoidal method T

$$\begin{aligned}
x_1 - x_0 &= \frac{s}{2}(a(x_1) + a(x_0)),\\
f^s - 1_N &= \frac{s}{2}(a \circ f^s + a),\\
f_T^s &= \Big(1_N - \frac{s}{2}a\Big)^{-1} \circ \Big(1_N + \frac{s}{2}a\Big) = f_I^{s/2} \circ f_E^{s/2}\\
&= (f_E^{s/2})^{-1} \circ f_C^s \circ f_E^{s/2} = 1_N + sa + \frac{s^2}{2}a^2 + \frac{s^3}{4}a^3 + O(s^4),\\
\log f_T^s &= sa + \frac{s^3}{12}a^3 + O(s^5),
\end{aligned}$$

revertible, ord $= 2$, symplectic for linear Hamiltonian but non-symplectic for non-linear Hamiltonian systems.

4. Centered Euler method C

$$\begin{aligned}
&x_1 - x_0 = sa\Big(\frac{1}{2}(x_1 + x_0)\Big),\\
&f^s - 1_N = sa \circ \Big(\frac{1}{2}(f^s + 1_N)\Big),
\end{aligned}$$

2-stage version recommended for implementation

$$\begin{aligned}
\bar{x} &= x + \frac{s}{2}a(\bar{x}), \qquad x_1 = 2\bar{x} - x_0,\\
\bar{x} &= f_I^{s/2}(x_0), \quad x_1 = 2f_I^{s/2}(x_0) - 1_N(x_0),\\
f_C^s &= 2f_I^{s/2} - 1_N = \Big(1_N + \frac{s}{2}a\Big)^{-1} \circ \Big(1_N - \frac{s}{2}a\Big) = f_E^{s/2} \circ f_I^{s/2}\\
&= 1_N + sa + \frac{s^2}{2}a^2 + \frac{s^3}{8}(a_*a^2 + a^3) + O(s^4),\\
\log f_C^s &= sa + s^3\Big(\frac{1}{8}a_*a^2 - \frac{1}{24}a^3\Big) + O(s^5),
\end{aligned}$$

revertible, ord $= 2$, unconditionally symplectic with preservation of all quadratic invariants for Hamiltonian systems.

Note the similarities and delicate differences between C and T: Both can be composed by a $s/2$ implicit and a $s/2$ explicit stages but in opposite orderings. Moreover, they are conjugate to each other, noted earlier independently in [7]. C is far less known than T, it becomes prominent only after the recent development of symplectic algorithms [3]. In crucial aspects C is superior.

Acknowledgements The author is grateful to Dr. Wang Dao-liu for discussions and valuable help in preparing the manuscript.

References

1 Feng K. The calculus of formal power series for diffeomorphisms and vector fields: [Preprint]. Beijing: CAS. Computing Center, 1991

2 Feng K. Formal dynamical systems and numerical algorithms. In Feng K, Shi Z C, ed. Proc Inter Conf on Computation of Differential Equations and Dynamical Systems. Singapore: World Scientific, 1993. 1–10

3 Feng K. On difference schemes and symplectic geometry. In Feng K, ed. Proc 1984 Beijing Symp Diff Geometry and Diff Equations. Beijing: Science Press, 1985. 42–58

4 Feng K. The Hamiltonian way for computing Hamiltonian dynamics. In Spigler R, ed. Applied and Industrial Mathematics. Netherlands: Kluwer, 1991. 17–35

5 Feng K, Qin M Z. Hamiltonian algorithms and a comparative numerical study. Comput Phys Comm. 1991 65: 173–187

6 Feng K. Symplectic, contact and volume-preserving algorithms. In Shi Z C, Ushijima T, ed. Proc 1st China-Japan Conf on Numer Math. Singapore: World Scientific, 1993. 1–28

7 Qin M Z. Oral communications