Drawing solution curve of differential equation

Farida Benmakrouha, Christiane Hespel, and Edouard Monnier

***Abstract*—We develop a method for drawing approximated solution curves of differential equations. This method is based on the juxtaposition of local approximating curves on successive intervals $[t_i, t_{i+1}]_{0 \le i \le n-1}$.**
The differential equation, considered as a dynamical system, is described by its state equations and its initial value at $t = t_0$.
A generic expression of its generating series G_t truncated at any order k, of the output and its derivatives $y^{(j)}(t)$ expanded at any order k, can be calculated. These expressions are obtained from the vector fields, from the observation of the state at time t, in the state equations [3], [7].
We get an expansion of $y^{(j)}(t)$ as a linear combination of differential monomials indexed by some colored partitions.
At every initial point of the present interval, we specify the previous expressions of G_t and $y^{(j)}(t)$ for $t = t_i$. Then we obtain an approximated output $y(t)$ at order k in every interval $[t_i, t_{i+1}]_{0 \le i \le n-1}$. We present an example from physics: the Duffing equation.
By using Maple system, we have developed a package corresponding to the creation of the generic expression of G_t and $y^{(j)}(t)$ at order k and to the drawing of the local curves on every interval $[t_i, t_{i+1}]_{0 \le i \le n-1}$, by iterations on the initial points $t = (t_i)_{0 \le i \le n-1}$.

***Index Terms*—analysis of dynamical systems, symbolic algorithm, generating series, colored partitions, rational approximation**

I. Introduction

The usual methods for drawing curves of differential equations consist in an iterative construction of isolated points, connected by straight lines (Runge-Kutta). Rather than calculate numerous successive approximate points $y(t_i)_{i \in I}$, it can be interesting to provide some few successive local curves $\{y(t)\}_{t \in [t_i, t_{i+1}]_{0 \le 1 \le n-1}}$.
Moreover, the computing of these local curves can be kept partly generic since a generic expression of the generating series G_{t_i} of the system can be provided in terms of t_i. The expression of the local curves $\{y(t)\}_{t \in [t_i, t_{i+1}]}$ is only a specification for $t = t_i$ at order k of the formula given in the proposition of section 3.
We consider a differential equation

$$y^{(N)}(t) = \phi(t, y(t), \cdots, y^{(N-1)}(t), u(t)) \qquad (1)$$

with initial conditions

$$y(0) = y_{0,0}, \cdots, y^{(N)}(0) = y_{0,N}$$

We assume that $\phi(t, y(t), \cdots, y^{(N-1)}(t), u(t))$ is polynomial in $y, \cdots, y^{(N-1)}$.
Then this differential equation can be viewed as an affine input $(u(t) = (u_j(t))_{1 \le j \le m})$ dynamical system.
By derivating, in the Fliess's formula, the expression of $y(t)$ in a neighborhood of $t = t_0$, we get an expansion of $y^{(n)}(t_0)$. This expression can be written as a linear combination of differential monomials $\otimes_{1 \le j \le m}(u_j^{(i_1)})^{e_1} \cdots (u_j^{(i_q)})^{e_q}$ indexed by some colored partitions $\mu = \otimes \mu_j$, for $u^\mu = u_1^{\mu_1} \cdots u_m^{\mu_m}$. And then there exist some polynomials g_μ in noncommutaive variables such that

$$y^{(n)}(t_0) = \sum_\mu \langle G_{t_0} | g_\mu \rangle \qquad (2)$$

For the partition $(\mu_j) = (u_j^{(i_1)})^{e_1} \cdots (u_j^{(i_q)})^{e_q}$, the weight $wgt(\mu_j)$ and the length $lg(\mu_j)$ are

$$\begin{array}{rcl} wgt(\mu_j) & = & \sum_{1 \le k \le m} e_k i_k \\ lg(\mu_j) & = & \sum_{1 \le k \le m} e_k \end{array} \qquad (3)$$

II. Preliminaries

A. *Affine system, Generating series*

We consider the nonlinear analytical system affine in the input:

$$(\Sigma) \qquad \left\{ \begin{array}{rcl} \dot{q} & = & f_0(q) + \sum_{j=1}^m f_j(q) u_j(t) \\ y(t) & = & g(q(t)) \end{array} \right. \qquad (4)$$

- $(f_j)_{0 \le j \le m}$ being some analytical vector fields in a neighborhood of $q(0)$
- g being the observation function analytical in a neighborhood of $q(0)$

Its initial state is $q(0)$ at $t = 0$. The generating series G_0 is built on the alphabet $Z = \{z_0, z_1, \cdots, z_m\}$, z_0 coding the drift and z_j coding the input $u_j(t)$. Generally G_0 is expressed as a formal sum $G_0 = \sum_{w \in Z^*} \langle G_0 | w \rangle w$ where $\langle G_0 | z_{j_0} \cdots z_{j_l} \rangle = f_{j_0} \cdots f_{j_l} g(q)|_{q(0)}$ depends on $q(0)$.

B. *Fliess's formula and iterated integrals*

The output $y(t)$ is given by the Fliess's equation ([3]):

$$y(t) = \sum_{w \in Z^*} \langle G_0 | w \rangle \int_0^t \delta(w) \qquad (5)$$

where G_0 is the generating series of (Σ) at $t = 0$:

$$\begin{array}{rcl} G_0 & = & \sum_{w \in Z^*} \langle G_0 | w \rangle w \\ & = & g(q)|_{q(0)} + \\ & & \sum_{l \ge 0} \sum_{j_i=0}^m f_{j_0} \cdots f_{j_l} g(q)|_{q(0)} z_{j_0} \cdots z_{j_l} \end{array} \qquad (6)$$

and $\int_0^t \delta(w)$ is the iterated integral associated with the word $w \in Z^* = \{z_0, z_1, \cdots, z_m\}^*$.

F. Benmakrouha, C. Hespel, E. Monnier are with the Department of Computer Science, INSA-IRISA, 20 avenue des Buttes de Coesmes, 35043 Rennes cedex, France, e-mail: Benma@insa-rennes.fr, Hespel@insa-rennes.fr, Monnier@insa-rennes.fr.

Manuscript received March 27, 2009.

Remember that the iterated integral $\int_0^t \delta(w)$ of the word w for the input u is defined by

$$\left\{\begin{array}{lcl} \int_0^t \delta(\epsilon) & = & 1 \\ \int_0^t \delta(vz_i) & = & \int_0^t \left(\int_0^\tau \delta(v)\right) u_i(\tau)d\tau \\ & & \forall z_i \in Z \quad \forall v \in Z^*. \end{array}\right. \tag{7}$$

where ϵ is the empty word, $u_0 \equiv 1$ is the drift and $u_i,\ 1 \le i \le m$ is the ith input.
We define the Chen's series as follows ([2])

$$C_u(t) = \sum_{w \in Z^*} \int_0^t \delta(w) \tag{8}$$

From the previous definitions, we obtain the following expression of $y(t)$

$$y(t) = \sum_{w \in Z^*} \langle G_0|w\rangle\langle C_u(t)|w\rangle \tag{9}$$

C. Iterated derivatives $y^{(n)}(0)$ of the output

G_0 being the generating series of the system, the ith derivative of $y(t)$ is

$$y^{(i)}(t) = \langle G_0|C_u^{(i)}(t)\rangle \tag{10}$$

We prove the following lemma ([6]) based on the Picart-Vessiot theory ([4])

Lemma :
Let be $\sum_{0\le j\le m} u_j.z_j = \mathrm{A}$. Then the derivative of the Chen's series is $\frac{d}{dt}C_u = C_u.\mathrm{A}$

From it, results the following recurrence relation:

$$C_u^{(i)} = C_u\mathrm{A_i}, \quad \mathrm{A_1} = \mathrm{A}, \quad \mathrm{A_{i+1}} = \mathrm{A}\mathrm{A_i} + D_t\mathrm{A_i} \tag{11}$$

D_t being the operator of time derivation.
Since $C_u(0) = 1$ and $C_u^{(i)}(0) = \mathrm{A_i}(0)$ then

$$y^{(i)}(0) = \sum_{w\in Z^*} \langle G_0|w\rangle\langle C_u^{(i)}(0)|w\rangle = \langle G_0|\mathrm{A_i}(0)\rangle \tag{12}$$

Let us remark that the successive derivatives $y(0), y^{(1)}(0), \cdots, y^{(k)}(0)$ are obtained from the coefficients $\langle G_0|w\rangle$ associated with the words whose length is $\le k$.
It results that the Taylor expansion of $y(t)$ up to order k only depends on the coefficients of G_0 truncated at order k.
For instance, for a single input $u(t)$ with drift $u_0(t) \equiv 1$, the derivatives are the following

$$\begin{array}{lcl} y(0) & = & \langle G_0|\epsilon\rangle \\ y^{(1)}(0) & = & \langle G_0|z_0\rangle + \langle G_0|z_1\rangle u(0) \\ y^{(2)}(0) & = & \langle G_0|z_0^2\rangle + (\langle G_0|z_0z_1\rangle + \langle G_0|z_1z_0\rangle)u(0) + \\ & & \langle G_0|z_1^2\rangle u(0)^2 + \langle G_0|z_1\rangle u^{(1)}(0) \\ \cdots & = & \cdots \end{array} \tag{13}$$

This method allows us to compute recursively the successive derivatives of $y(t)$ at $t = 0$.
The derivation law D of the partitions, producing the effect of the time derivation D_t of the differential monomials satisfies

$$\begin{array}{lcl} D(i_k) & = & i_{k+1} \\ D(i_1^{e_1}\cdots i_q^{e_q}) & = & \sum_{k=1}^q e_k \times (i_1^{e_1}\cdots i_k^{e_k-1} i_{k+1}^{e_{k+1}+1}\cdots i_q^{e_q}) \end{array} \tag{14}$$

For a single input $u(t)$ with drift $u_0(t) \equiv 1$, the bicolored multiplicity is $\mu = \mu_0 \otimes \nu$ with

$$\begin{array}{lcl} \mu_0 & = & 1^p \\ wgt(\mu) & = & p + wgt(\nu) \\ D(1^p \otimes \nu) & = & 1^p \otimes D(\nu) \end{array} \tag{15}$$

III. Approximate value of $y^{(n)}(t)$

The Fliess's formula can be written

$$y(t) = \langle G_0|\epsilon\rangle + \sum_{w\in Z^*-\{\epsilon\}} \langle G_0|w\rangle\langle C_u(t)|w\rangle \tag{16}$$

An approximate function $y_k(t)$ de $y(t)$ up to order k in a neighborhood of $t = 0$ is obtained by expanding this expression up to the same order k. Then we have

$$|y(t) - y_k(t)| = O(t^{k+1}) \tag{17}$$

For instance, at order $k = 1$, $y(t)$ has the following approximate expression for a single input with drift

$$y_1(t) = \langle G_0|\epsilon\rangle + \langle G_0|z_0\rangle t + \langle G_0|z_1\rangle\xi_1(t) \tag{18}$$

where $\xi_k(t)$ denotes the kth primitive of $u(t)$.
This computing can be generalized to the successive derivatives of $y(t)$.

Proposition
Given the expression of $y^{(n)}(0)$ in terms of the coefficients of G_0 and of the derivatives of order $\le n-1$ of the input $u(t)_{t=0}$ obtained recursively according to the previous section, we can deduce the expression of $y^{(n)}(t)$ by executing in $y^{(n)}(0)$ the following transformations

1) **We substitute $u^{(i)}(t)$ to $u^{(i)}(0)$ for $0 \le i \le n-1$**
2) **For every occurrence of a coefficient $\langle G_0|v\rangle$ where $v \in Z^*$, we add the following corrective term**

$$\sum_{w\ne\epsilon} \langle G_0|wv\rangle\langle C_u(t)|w\rangle$$

The proof is based on the following properties

$$\left\{\begin{array}{lcl} \frac{d}{dt}\langle C_u(t)|vz_i\rangle & = & \langle C_u(t)|v\rangle u_i(t) \\ \langle C_u(t)|\epsilon\rangle & = & 1 \end{array}\right. \tag{19}$$

For instance, for a single input with drift, we compute from

$$y^{(1)}(0) = \langle G_0|z_0\rangle + \langle G_0|z_1\rangle u(0)$$

the expression of $y^{(1)}(t)$:

$$\begin{array}{lcl} y^{(1)}(t) & = & \langle G_0|z_0\rangle + \sum_{w\ne\epsilon}\langle G_0|wz_0\rangle\langle C_u(t)|w\rangle + \\ & & (\langle G_0|z_1\rangle + \sum_{w\ne\epsilon}\langle G_0|wz_1\rangle\langle C_u(t)|w\rangle)u(t) \end{array} \tag{20}$$

By restricting the sums to the words w whose length $|w|$ satisfies $1 \le |w| \le k$, we obtain a function $y_k^{(n)}(t)$ approximating $y^{(n)}(t)$ up to order k. And then

$$|y_k^{(n)}(t) - y^{(n)}(t)| = O(t^{k+1}) \tag{21}$$

A. Generalization at time $t = t_i$

For a single input with drift, the system (Σ) can be written at $t = t_i$:

$$\left\{\begin{array}{lcl} \dot{q}(t_i+h) & = & f_0(q(t_i+h)) + f_1(q(t_i+h))u(t_i+h) \\ y(t_i+h) & = & g(q(t_i+h)) \end{array}\right. \tag{22}$$

By setting

$$\left\{\begin{array}{lcl} U_i(h) & = & u(t_i+h) \\ Y_i(h) & = & y(t_i+h) \\ Q_i(h) & = & q(t_i+h) \end{array}\right. \tag{23}$$

we obtain the following system

$$(\Sigma_i) \qquad \left\{\begin{array}{lcl} \dot{Q}_i(h) & = & f_0(Q_i(h)) + f_1(Q_i(h)U_i(h) \\ Y_i(h) & = & g(Q_i(h)) \end{array}\right. \tag{24}$$

And G_i is the generating series of (Σ_i).
By setting $\psi_{i,k}(h) = \xi_k(t_i+h)$, then $\psi_{i,k}(h)$ is the kth primitive of $u(t_i+h)$ or the kth primitive of $U_i(h)$.

We have the equalities

$$\xi_1(t_i+h) = \int_{t_i}^{t_i+h} u(\tau)d\tau = \int_0^h U_i(t)dt = \psi_{i,1}(h) \tag{25}$$

And then, we can prove recursively that the Chen's integral $\int_{t_i}^{t_i+h}\delta(w)$ can be computed as an integral $\int_0^t \delta(W)$ by considering $U_i(t)$ instead of $u(t_i+t)$.

IV. Application to curves drawing

We present an application to the curve drawing of the solution of differential equations. We consider a differential equation

$$y^{(N)}(t) = \phi(t, y(t), \cdots, y^{(N-1)}(t), u(t)) \tag{26}$$

with initial conditions

$$y(0) = y_{0,0}, \cdots, y^{(N)}(0) = y_{0,N}$$

It can be written for $y = q_1$:

$$\left\{\begin{array}{lcl} q_1^{(1)} & = & q_2 \\ q_2^{(1)} & = & q_3 \\ \cdots & = & \cdots \\ q_N^{(1)} & = & \phi(t, q_1, \cdots, q_N) \end{array}\right. \tag{27}$$

We assume that

$$\phi(t, q_1, \cdots, q_N) = P_0(q_1, \cdots, q_N) + \sum_{j=1}^{m} P_j(q_1, \cdots, q_N)u_j(t)$$

for $P_0, P_1, \cdots, P_m$ polynomials in commutative variables $q_1, \cdots, q_N$.

For an analytical affine single input system (Σ) then $m = 1$ and the vector fields are $f_0,\ \ f_1$, corresponding to P_0, P_1.

We propose a curve drawing of the output $y(t)$ of this system in $[0, T] = \bigcup[t_i, t_{i+1}]_{0\le i\le n-1}$ according to the following algorithm:
Firstly, we compute a generic expression of the generating series G_t.

- Initial point $t_0 = 0$:
  $y(0) = q_1(0), \cdots, y^{(N-1)}(0) = q_N(0)$ are given.
  The vector fields f_0, f_1 applied to $g(q)$ evaluated in t_0 provide $\langle G_0|w\rangle$ for $|w| \le k$
- Step i:
  Knowing $y(t_{i-1}) = q_1(t_{i-1}), \cdots, y^{(N-1)}(t_{i-1}) = q_N(t_{i-1})$ and $\langle G_{i-1}|w\rangle$ (for $|w| \le k$), we compute $y(t_i), \cdots, y^{(N-1)}(t_i)$ according to section 3 and $\langle G_i|w\rangle$ (for $|w| \le k$) by applying the vector fields f_0, f_1 to $g(q)$ at $q(t_i)$.
  We draw the local curve of the function $t_{i-1} + dt \to y(t_{i-1} + dt)$ on the interval $[t_{i-1}, t_i]$.
- Final point $t = T = t_n$:
  stop at $i = n$.

A. Genericity of the method

The computing of the coefficients

$$\langle G_i|z_{j_0}\cdots z_{j_l}\rangle = f_{j_0}\cdots f_{j_l}g(q)|_{q(t_i)}$$

is generic.
The computing of the expressions of

$$Y_i(h) = y(t_i+h) = y(t_i) + \sum_{|w|\le k}\langle G_i|w\rangle\langle C_{U_i}(h)|w\rangle$$

and of

$$\begin{array}{lcl} Y_i^{(1)}(h) & = & \langle G_i|z_0\rangle + \sum_{1\le|w|\le k}\langle G_i|wz_0\rangle\langle C_{U_i}(h)|w\rangle + \\ & & (\langle G_i|z_1\rangle + \sum_{1\le|w|\le k}\langle G_i|wz_1\rangle\langle C_{U_i}(h)|w\rangle)U_i(h) \end{array} \tag{28}$$

are generic too.
We use the previous algorithm by specifying t_i at every step in the previous expressions.

B. Example: Duffing equation

Its equation is the following:

$$\begin{array}{lcl} y^{(2)}(t) + ay^{(1)}(t) + by(t) + cy^3(t) & = & u(t) \\ y(0) = y_0, & & \\ y^{(1)}(0) = y_{1,0} & & \end{array} \tag{29}$$

It can be written as a first order differential system

$$\left\{\begin{array}{lcl} q_1^{(1)}(t) & = & q_2(t) \\ q_2^{(1)}(t) & = & -aq_2(t) - bq_1(t) - cq_1^3(t) + u(t) \\ & = & F(q(t)) + u(t) \\ y(t) & = & q_1(t) = g(q) \\ q_1(0) = y_0, & & q_{2,0} = y_{1,0} \end{array}\right. \tag{30}$$

The vector fields are

$$\begin{array}{lcl} f_0(q_1, q_2) & = & q_2\frac{\partial}{\partial q_1} - (aq_2 + bq_1 + cq_1^3)\frac{\partial}{\partial q_2} \\ & = & q_2\frac{\partial}{\partial q_1} + F(q)\frac{\partial}{\partial q_2} \end{array}$$

$$f_1(q_1, q_2) = \frac{\partial}{\partial q_2}$$

1) We write generic equations describing the generating series G_i at $t = t_i$:

$$\forall t_i \quad \langle G_i | z_{j_1} \cdots z_{j_l} \rangle = (f_{j_1} \cdots f_{j_l} g(q))|_{q(t_i)}$$

Let us remark that

$$\langle G_i | w z_1 \rangle = 0 \ \ \forall w \in Z^*, \quad \langle G_i | w z_1 z_0 \rangle = 0 \ \ \forall w \in Z^+$$

For instance, for order $k = 3$, we have only to compute 6 coefficients of G_i instead of 15 coefficients.

$$\begin{array}{lcl} \langle G_i | \epsilon \rangle & = & q_1(t_i) \\ \langle G_i | z_0 \rangle & = & q_2(t_i) \\ \langle G_i | z_0^2 \rangle & = & F(q(t_i)) \\ \langle G_i | z_1 z_0 \rangle & = & 1 \\ \langle G_i | z_0^3 \rangle & = & (q_2 \frac{\partial}{\partial q_1} F(q) + F(q) \frac{\partial}{\partial q_2} F(q))_{q(t_i)} \\ \langle G_i | z_1 z_0^2 \rangle & = & -a \end{array} \tag{31}$$

2) We write generic approximate expression of the output $y(t_{i+1})$ and its derivative $y^{(1)}(t_{i+1})$ for every $t = t_{i+1} = t_i + h$ at order k:

$$\begin{array}{l} y(t_{i+1}) = \langle G_i | \epsilon \rangle + \sum_{1 \leq |w| \leq k} \langle G_i | w \rangle \langle C_{U_i}(h) | w \rangle \\ y^{(1)}(t_{i+1}) = \langle G_i | z_0 \rangle + \\ \sum_{1 \leq |w| \leq k} \langle G_i | w z_0 \rangle \langle C_{U_i}(h) | w \rangle + \\ (\langle G_i | z_1 \rangle + \sum_{1 \leq |w| \leq k} \langle G_i | w z_1 \rangle \langle C_{U_i}(h) | w \rangle) U_i(h) \end{array} \tag{32}$$

For instance, for order $k = 3$

$$\begin{array}{lcl} Y_i(h) & = & y(t_i + h) \\ & = & y(t_i) + \langle G_i | z_0 \rangle h + \langle G_i | z_0^2 \rangle h^2/2 + \\ & & \langle G_i | z_1 z_0 \rangle \psi_{i,2}(h) + \langle G_i | z_0^3 \rangle h^3/(3!) + \\ & & \langle G_i | z_1 z_0^2 \rangle \psi_{i,3}(h) \end{array} \tag{33}$$

and

$$\begin{array}{lcl} Y_i^{(1)}(h) & = & y^{(1)}(t_i + h) \\ & = & \langle G_i | z_0 \rangle + \langle G_i | z_0^2 \rangle h + \\ & & \langle G_i | z_1 z_0 \rangle \psi_{i,1}(h) + \langle G_i | z_0^3 \rangle h^2/2 + \\ & & \langle G_i | z_1 z_0^2 \rangle \psi_{i,2}(h) \end{array} \tag{34}$$

3) And we use the algorithm of section 4 by specifying t_i at every step. So we obtain the drawing of $y(t)$.

C. Contribution of symbolic computing

The symbolic computing allows us to profit from the genericity and from the precision.

1) Genericity : We propose that one uses the formal expression of the generating series G_i and of the output $y(t_i)$ and its derivative $y^{(1)}(t_i)$. Then we replace successively the expressions by their values at every step.
2) Precision : We can choose any order k for approximating the output and its derivative. The error is on the order of $k + 1$.

D. Comparison with other methods

The main interest of this method consists in choosing the precision, not only by the size of the time interval h but by the order of the approximation.

The quality of any approximation depends on the order, the size of the interval but also depends on the roughness of the curve and the stability of the system [1]. When the system is stable, the drawing of the curve is suitable, by using our method, for a large period of time and a small order. In this case, our method is favourable. Otherwise we have to reduce the period of time in order to follow the true curve.

In comparison with Runge-Kutta methods, this method consists in selecting a much smaller number of steps, the local curve being acquired on every interval.

REFERENCES

[1] Benmakrouha F., Hespel C., *Generating series for the study of stability of bilinear systems*, 23rd IFIP TC7 Conference on System Modelling and Optimization, Cracow, july 2007.

[2] Chen K.T., *Iterated path integrals*, Bull. Amer. Math. Soc. 83, pp. 831-879, 1977.

[3] Fliess M., *Fonctionnelles causales non linéaires et indéterminés non commutatives*, Bull. Soc. Math. France 109, pp. 3-40, 1981.

[4] Fliess M., Reutenauer C., *Théorie de Picard-Vessiot des systèmes réguliers*, Colloque Nat.CNRS-RCP 567, Outils et Modèles Mathématiques pour l'Automatique,l'Analyse des Systèmes et le Traitement du Signal, Belle-Ile, sept. 1982.

[5] Fliess M., Lamnabhi M., Lamnabhi-Lagarrigue F., *An algebraic approach to nonlinear functional expansions*, IEEE Trans. Circuits and Systems CAS-30, pp. 554-570, 1983.

[6] Hespel C., *Iterated derivatives of a nonlinear dynamic system and Faà di Bruno formula*, Mathematics and Computers in Simulation, vol.42, pp. 641-657, 1996.

[7] Hespel C., *Une étude des séries formelles non commutatives pour l'Approximation et l'Identification des systèmes dynamiques*, Thèse d'état, Université de Lille 1, 1998.