Discrete-time Chen Series for Time Discretization and Machine Learning

W. Steven Gray
Department of Electrical and Computer Engineering
Old Dominion University
Norfolk, VA 23529 USA

G. S. Venkatesh
Department of Electrical and Computer Engineering
Old Dominion University
Norfolk, VA 23529 USA

Luis A. Duffaut Espinosa
Department of Electrical and Biomedical Engineering
University of Vermont
Burlington, VT 05405 USA

***Abstract*—A formal power series over a set of noncommuting indeterminants using iterated integrals as the coefficients is called a *Chen series*, named after the mathematician K.-T. Chen. The first goal of this paper is to give a brief overview of Chen series and their algebraic structures as a kind of reference point. The second goal is to describe its discrete-time analogue in detail and then apply the concept in two problems, the time discretization problem for nonlinear control systems and the machine learning problem for dynamical systems.**

Keywords*—*nonlinear control systems, machine learning, discretization, formal power series

I. Introduction

Given a piecewise continuous function $u : [t_0, t_1] \to \mathbb{R}^m$, consider the set of iterated integrals computed from line integrals along the various m coordinate axes in specific orders. A formal power series over a set of m noncommuting indeterminants X using these iterated integrals as the coefficients is called a *Chen series*, named after the mathematician K.-T. Chen who first characterized their underlying algebraic properties in a series of papers beginning with [2]. The concept was later adapted by Fliess in [5] and further enhanced by Sussmann in [16] for the purpose of representing the input-output map of a nonlinear control in terms of a weighted sum of iterated integrals, the so called *Chen-Fliess series* or *Fliess operator*. Subsequently, the Chen series was recast in the context of a tensor algebra over X and referred to as the *signature* in order to describe the solution of a differential equation driven by a *rough path* [14]. More recently, a discrete-time notion of a Chen series was proposed in [3], [7] for the purpose of numerical approximation of Fliess operators. The idea in those works was simply to replace the iterated integrals with iterated sums and to provide explicit bounds on the discretization error. The corresponding notion of a discrete-time Fliess operator turned out to be very convenient for solving data-driven/model-free control problems [8]. Specifically, input-output data from some unknown plant, assumed to have a Fliess operator representation, is used to identify the generating series coefficients. The discrete-time Fliess operator approximator is then used for predictive control. This type of generic system identification for dynamical systems is now evolving into an alternative notion of machine learning as networks of discrete-time Fliess operators are interconnected as learning units in order to achieve a control objective [9].

The goals of this paper are two-fold. First a brief overview of Chen series and their algebraic structures is provided as a kind of reference point. Then their discrete-time analogue is presented in more algebraic detail than was done in [3], [7]. Next, these ideas are applied to two related problems, the time discretization problem for nonlinear control systems and the machine learning problem for dynamical systems. Perhaps the latter is the most interesting as the underlying algebraic and combinatorial structures need to be robustly exercised in order to provide efficient learning algorithms, especially for multivariable problems where the combinatorial complexity grows rapidly. Some suggestions for future work in this regard are outlined.

II. Chen Series

A finite nonempty set of noncommuting symbols $X = \{x_0, x_1, \ldots, x_m\}$ is called an *alphabet*. A *word* $\eta = x_{i_1} \cdots x_{i_k}$ over X is any finite sequence of *letters* from X. Its *length* is $|\eta| = k$. The set of all words of length k is denoted by X^k, while X^* is the set of all words, including the empty word $\emptyset$. It is immediate that X^* is a monoid under catenation. (That is, a set with an associative product and an identity element.) Any mapping $c : X^* \to \mathbb{R}^\ell$ is called a *formal power series*. The value of c at $\eta \in X^*$ is denoted by (c, η) and called the *coefficient* of η in c. A series c is *proper* when $(c, \emptyset) = 0$. The *support* of c, $\mathrm{supp}(c)$, is the set of all words having nonzero coefficients. Normally, c is written as a formal sum $c = \sum_{\eta \in X^*} (c, \eta)\eta$. The collection of all formal power series over X is denoted by $\mathbb{R}^\ell\langle\langle X\rangle\rangle$. It constitutes a noncommutative associative $\mathbb{R}$-algebra under the catenation (Cauchy) product. A series $c \in \mathbb{R}\langle\langle X\rangle\rangle$ is called a *Lie series* if it can be decomposed as $c = \sum_{n\geq 1} p_n$, where each p_n is a polynomial in the free Lie algebra $\mathcal{L}(X)$ over X with support residing X^n [15]. The series c is called an *exponential Lie series* when $c = \exp(d) := \sum_{n\geq 0} d^n/n!$, where d is a Lie series. An important class of exponential Lie series are Chen series as described below.

Definition 2.1: [2], [16] Let $X = \{x_0, x_1, \ldots, x_m\}$. For any fixed $u \in L_1^m[t_0, t_1]$ and $t \in [t_0, t_1]$ one can associate the formal power series in $\mathbb{R}\langle\langle X\rangle\rangle$

$$P[u](t, t_0) = \sum_{\eta \in X^*} \eta\, E_\eta[u](t, t_0),$$

where for each word $\eta \in X^*$ the map $E_\eta : L_1^m[t_0, t_1] \to C[t_0, t_1]$ is defined inductively by setting $E_\emptyset[u] = 1$ and letting

$$E_{x_i\bar{\eta}}[u](t, t_0) = \int_{t_0}^{t} u_i(\tau) E_{\bar{\eta}}[u](\tau, t_0)\, d\tau,$$

978-1-7281-1151-3/19/$31.00

with $x_i \in X$, $\bar{\eta} \in X^*$, and $u_0(t) := 1$. Such a series is called a ***Chen series***. When $t_0 = 0$, $P[u](t, t_0)$ will be abbreviated as $P[u](t)$.

Example 2.1: Suppose $X = \{x_0, x_1\}$ and $u(t) = \hat{u} \in \mathbb{R}$ on $[0, T]$. It follows directly that $P[u](0) = 1$ and

$$\begin{aligned} \frac{d}{dt}P[u](t) &= \sum_{\eta \in X^*} \eta \frac{d}{dt} E_\eta[u](t,0) \\ &= \sum_{\eta \in X^*} x_0 \eta\, E_\eta[u](t,0) + x_1 \eta\, \hat{u} E_\eta[u](t,0) \\ &= (x_0 + x_1 \hat{u}) P[u](t). \end{aligned} \tag{1}$$

Similarly,

$$\frac{d^n}{dt^n} P[u](0) = (x_0 + x_1 \hat{u})^n, \ \ n \geq 0,$$

and, therefore,

$$\begin{aligned} P[u](t) &= \sum_{n=0}^{\infty} (x_0 + x_1 \hat{u})^n \frac{t^n}{n!} \\ &= \mathrm{e}^{(x_0 + x_1 \hat{u})t} \end{aligned}$$

is the solution of the formal differential equation (1). Clearly, $(x_0 + x_1 \hat{u})t \in \mathcal{L}(X)$. □

The following theorem generalizes the previous example.

Theorem 2.1: [16] Let $X = \{x_0, x_1, \ldots, x_m\}$. For any finite $T > 0$ let $u \in L_1^m[0, T]$. Then the Chen series $P[u](t)$ is an exponential Lie series for every $t \in [0, T]$, and

$$\frac{d}{dt} P[u] = \left[x_0 + \sum_{i=1}^{m} x_i u_i \right] P[u], \ \ P[u](0) = 1.$$

The notion of a Chen series is implicit in the definition of a Fliess operator $y = F_c[u]$ with generating series $c \in \mathbb{R}\langle\langle X \rangle\rangle$. Namely, for any $t \geq t_0$

$$\begin{aligned} y(t) &:= \sum_{\eta \in X^*} (c, \eta) E_\eta[u](t, t_0) \\ &= \sum_{\eta \in X^*} (c, \eta)(P[u](t, t_0), \eta) \\ &= (c, P[u](t, t_0)), \end{aligned}$$

where $(\cdot, \cdot)$ in the last line is viewed as a scalar product on $\mathbb{R}\langle\langle X \rangle\rangle \times \mathbb{R}\langle\langle X \rangle\rangle$, and all the underlying series are assumed to converge in some manner [10]. In this context, the letter x_0 is used to represent nonhomogeneous operators, i.e., those for which $F_c[0] \neq 0$.

Example 2.2: Let $c \in \mathbb{R}\langle\langle X \rangle\rangle$ and $u \in L_1^m[0, T]$. Since $P[u](t)$ is in general an exponential Lie series, it follows that the Fliess operator corresponding to c can always be written in the form $y(t) = F_c[u](t) = (c, \mathrm{e}^{U(t)})$, where $U(t) = \ln(P[u](t)) := \sum_{k \geq 1} ((-1)^{k+1}/k)(P[u](t) - 1)^k$, $t \in [0, T]$. □

A central property of Chen series is that they are closed under the Cauchy product. Consider two input functions $(u, v) \in L_1^m[t_a, t_b] \times L_1^m[t_c, t_d]$. The *durations* of u and v are taken to be $t_b - t_a \geq 0$ and $t_d - t_c \geq 0$, respectively, and the functions are not defined outside their corresponding

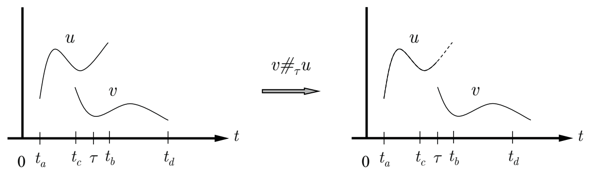

Fig. 1. The catenation of two inputs u and v at $t = \tau$.

intervals. The *catenation* of u and v at $\tau \in [t_a, t_b]$ is taken to be

$$(v \#_\tau u)(t) = \begin{cases} u(t) : t_a \leq t \leq \tau \\ v((t - \tau) + t_c) : \tau < t \leq \tau + (t_d - t_c), \end{cases}$$

as shown in Figure 1. It is easily verified that the set of functions

$$L_1^m(0) := \bigcup_{0 \leq T < \infty} L_1^m[0, T]$$

is a monoid under this catenation operator. The identity element in this case is denoted by $\mathbf{0}$ and is equivalent to the set of functions having exactly zero duration (i.e., $T = 0$). By definition $P[\mathbf{0}] = 1$.

Theorem 2.2: (Chen's identity), [2], [16] Given $(u, v) \in L_1^m[t_a, t_b] \times L_1^m[t_c, t_d]$, $\tau \in [t_a, t_b]$, and $t \in [\tau, \tau + (t_d - t_c)]$ it follows that

$$P[v]((t - \tau) + t_c, t_c) P[u](\tau, t_a) = P[v \#_\tau u](t, t_a).$$

For the special case where $t_c = \tau$, Chen's identity reduces to

$$P[v](t, \tau) P[u](\tau, t_a) = P[v \#_\tau u](t, t_a).$$

Another useful special case is when $t_a = t_c = 0$ so that

$$P[v](t - \tau) P[u](\tau) = P[v \#_\tau u]((t - \tau) + \tau).$$

Since $P[u](0) = 1$ for any fixed $u \in L_1[0, T]$, the set of Chen series

$$\begin{aligned} \mathcal{G}_C(X) = \{ & P[u](t) \in \mathbb{R}\langle\langle X \rangle\rangle : u \in L_1^m[0, T], \\ & 0 \leq t \leq T < \infty \} \end{aligned}$$

defines a monoid under the Cauchy product. In fact, the mapping $P : L_1^m(0) \to \mathcal{G}_C(X)$ acts as a monoid homomorphism.

Example 2.3: Consider an alphabet for a two input system with no letter x_0, i.e., $X = \{x_1, x_2\}$. Assume $u(t) = [\hat{u}(0) \ \ 0]^T$ on $[0, T]$ and $v(t) = [0 \ \ \hat{u}(1)]^T$ on $[T, 2T]$, where $\hat{u}(i) \in \mathbb{R}$, $i = 1, 2$. Then for any $t \in [T, 2T]$ the Chen series for $v \#_T u$ is

$$\begin{aligned} P[v \#_T u](t, 0) &= P[v](t, T) P[u](T, 0) \\ &= \sum_{j,k=0}^{\infty} x_2^k x_1^j \, E_{x_2^k}[v](t, T) E_{x_1^j}[u](T, 0) \\ &= \sum_{k=0}^{\infty} (\hat{u}(1)(t - T)x_2)^k \frac{1}{k!} \sum_{j=0}^{\infty} (\hat{u}(0) T x_1)^j \frac{1}{j!} \\ &= \mathrm{e}^{\hat{u}(1)(t-T)x_2} \mathrm{e}^{\hat{u}(0) T x_1}. \end{aligned}$$

From the Campbell-Baker-Hausdorff formula it follows directly that $P[v \#_T u](t, 0)$ is an exponential Lie series as asserted in Theorem 2.1. □

Example 2.4: Reconsider Example 2.3 with $v = -u$ and $t = 2T$. In which case,

$$P[(-u)\#_T u](2T,0) = \mathrm{e}^{-\hat{u}(0)Tx_1}\mathrm{e}^{\hat{u}(0)Tx_1} = 1,$$

or equivalently,

$$P[-u](2T,T)P[u](T,0) = 1.$$

Note that if the *drift letter* x_0 is admitted, then $P[0](T,0) = \mathrm{e}^{Tx_0}$. In this case, there is no input available to generate a Chen series corresponding to the inverse series e^{-Tx_0}. □

As suggested by the previous example, the final theorem of this section shows that $\mathcal{G}_C(X)$ constitutes a group if the letter x_0 is omitted. The proof is straightforward in terms of the *shuffle product* on $\mathbb{R}\langle\langle X\rangle\rangle$, namely, the bilinear product defined inductively on words as

$$(x_i\eta) ⧢ (x_j\xi) = x_i(\eta ⧢ (x_j\xi)) + x_j((x_i\eta) ⧢ \xi),$$

where $x_i, x_j \in X$, $\eta, \xi \in X^*$ and with $\eta ⧢ \emptyset = \emptyset ⧢ \eta = \eta$ [5].

Theorem 2.3: [2], [16] Let $X = \{x_1, \ldots, x_m\}$. The set of Chen series $\mathcal{G}_C(X)$ is a group under the Cauchy product.

Proof: Since $\mathcal{G}_C(X)$ is a monoid under the Cauchy product, it only remains to be shown that every element in $\mathcal{G}_C(X)$ has a well defined inverse. For any given $u \in L_1^m[0,T]$ and fixed $t \in [0,T]$ define the input functions $u_{S,i}(\tau) = -u_i(t-\tau)$ restricted to $[0,t]$ for $i = 1, 2, \ldots, m$. (Note that this is not possible for $u_0 = 1$.) It is straightforward to show for any $\xi \in X^*$ that $E_\xi[u_S](\tau,0) = E_{S(\xi)}[u](\tau,0)$ for all $\tau \in [0,t]$, where $S(\xi) := (-1)^{|\xi|}\tilde{\xi}$, and $\tilde{\xi}$ denotes the word ξ with the letters written in reverse order. In which case, for $P[u](t) \in \mathcal{G}_C(X)$

$$\begin{aligned} P[u](t)P[u_S](t) &= \sum_{\eta,\xi\in X^*} \eta\xi\, E_\eta[u](t,0)E_\xi[u_S](t,0) \\ &= \sum_{\eta,\xi\in X^*} (-1)^{|\xi|}\eta\xi\, E_{\eta ⧢ \tilde{\xi}}[u](t,0). \end{aligned}$$

Given any $\nu \in X^*$, it then follows that

$$\begin{aligned} (P[u](t)P[u_S](t),\nu) &= \sum_{\eta,\xi\in X^*} (-1)^{|\xi|}(\eta\xi,\nu)\, E_{\eta ⧢ \tilde{\xi}}[u](t,0) \\ &= \begin{cases} 1 & : \ |\nu| = 0 \\ 0 & : \ |\nu| > 0, \end{cases} \end{aligned}$$

where the identity

$$\sum_{\eta,\xi\in X^*} (-1)^{|\xi|}\,(\eta\xi,\nu)\,\eta ⧢ \tilde{\xi} = \begin{cases} 1 & : \ |\nu| = 0 \\ 0 & : \ |\nu| > 0 \end{cases}$$

has been used. Thus, $P[u](t)P[u_S](t) = 1$. Similarly, one can show that $P[u_S](t)P[u](t) = 1$, and thus, $(P[u](t))^{-1} = P[u_S](t) \in \mathcal{G}_C(X)$.

Finally, it should be noted that in Chen's original work, the concept of a *path* was used instead of an *input*. A path $U : [0,1] \to \mathbb{R}^m$ corresponding to an input $u \in L_1^m[0,1]$ is defined to have component functions

$$U_i(t) = \int_0^t u_i(\tau)\,d\tau, \quad i = 1, 2, \ldots, m.$$

The catenation of paths V and U includes a renormalization using arc length so that $V\#_1 U$ always defines another path on $[0,1]$. In this setting, U^{-1} is the path associated with the input u_S as defined in the proof of Theorem 2.3. In particular, $(U^{-1})^{-1} = U$, and $U^{-1}\#_1 U$ is the path from $U(0)$ to $U(1)$ and then back to $U(0)$ retracing the first path but in the opposite direction. Modulo such *null paths*, Chen shows that the set of all paths on $[0,1]$ forms a group $\mathcal{G}_P$ so that the corresponding Chen series map $P : \mathcal{G}_P \mapsto \mathcal{G}_C(X)$ taking paths (rather than inputs) to the Chen group is a *group homomorphism*. In the present setting, however, the notion of a path is too specialized.

III. Discrete-time Chen Series

In the discrete-time setting, inputs are sequences of vectors from the normed linear space

$$l_\infty^{m+1}(N_0) := \{\hat{u} = (\hat{u}(N_0), \hat{u}(N_0+1), \ldots) : \|\hat{u}\|_\infty < \infty\},$$

where $\hat{u}(N) := [\hat{u}_0(N), \hat{u}_1(N), \ldots, \hat{u}_m(N)]^T$, $N \geq N_0$ with $\hat{u}_i(N) \in \mathbb{R}$, $|\hat{u}(N)| := \max_{i\in\{0,1,\ldots,m\}} |\hat{u}_i(N)|$, and $\|\hat{u}\|_\infty := \sup_{N\geq N_0} |\hat{u}(N)|$. The subset of finite sequences over $[N_0, N_f]$ is denoted by $l_\infty^{m+1}[N_0, N_f]$. In contrast to the continuous-time case, it is more convenient to explicitly include the drift input $\hat{u}_0$, which is constant but not necessarily unity, as part of the input $\hat{u}$ [7], [8].

Definition 3.1: [7] Given any $N \geq N_0$ and $\hat{u} \in l_\infty^{m+1}(N_0)$, a **discrete-time Chen series** is defined as

$$S[\hat{u}](N,N_0) = \sum_{\eta\in X^*} \eta S_\eta[\hat{u}](N,N_0),$$

where

$$S_{x_i\eta}[\hat{u}](N,N_0) = \sum_{k=N_0}^{N} \hat{u}_i(k) S_\eta[\hat{u}](k,N_0) \tag{2}$$

with $x_i \in X$, $\eta \in X^*$, and $S_\emptyset[\hat{u}](N,N_0) := 1$. If $N_0 = 0$ then $S[\hat{u}](N,0)$ is abbreviated as $S[\hat{u}](N)$.

Example 3.1: Let X be arbitrary and define $\hat{u}_\eta(N) = \hat{u}_{i_k}(N)\cdots\hat{u}_{i_1}(N)$ for any $\eta = x_{i_k}\cdots x_{i_1} \in X^*$ and $N \geq N_0$ with $\hat{u}_\emptyset(N) := 1$. In addition, $c_u(N) := \sum_{\eta\in X^*} \hat{u}_\eta(N)\eta$. Then

$$S_{x_i\eta}[\hat{u}](N_0,N_0) = \hat{u}_{x_i}(N_0) S_\eta[\hat{u}](N_0,N_0)$$

so that $S_\eta[\hat{u}](N_0,N_0) = \hat{u}_\eta(N_0)$, and thus, $S[\hat{u}](N_0,N_0) = c_u(N_0)$. For example, if $X = \{x_1\}$ and $\hat{u}_{x_1}(N_0) = \hat{u}_1(N_0)$, then $S[\hat{u}](N_0,N_0) = \sum_{k\geq 0}(\hat{u}_1(N_0)x_1)^k =: (1 - \hat{u}_1(N_0)x_1)^{-1}$. □

Analogous to the continuous-time case, the discrete-time Chen series $S[\hat{u}](N,N_0)$ satisfies a difference equation as described next. The following theorem is a generalization of that appearing in [8].

Theorem 3.1: For any $\hat{u} \in l_\infty^{m+1}(N_0)$, $\eta \in X^*$ and $N \geq N_0$

$$S[\hat{u}](N+1,N_0) = c_u(N+1) S[\hat{u}](N,N_0)$$

with $S[\hat{u}](N_0,N_0) = c_u(N_0)$ so that

$$S[\hat{u}](N,N_0) = \overleftarrow{\prod_{i=N_0}^{N}} c_u(i), \tag{3}$$

where $\overleftarrow{\prod}$ denotes a directed product from right to left.

Proof: The first identity is addressed by proving that

$$S_\eta[\hat{u}](N+1,N_0) = (c_u(N+1)S[\hat{u}](N,N_0),\eta), \quad \forall \eta \in X^*$$

via induction on the length of η. When $\eta = \emptyset$ then trivially $S_\emptyset[\hat{u}](N+1,N_0) = 1 = \hat{u}_\emptyset(N+1)S_\emptyset[\hat{u}](N,N_0)$. If $\eta = x_i \in X$ then from (2)

$$\begin{aligned} S_{x_i}[\hat{u}](N+1,N_0) &= \hat{u}_{x_i}(N+1) + S_{x_i}[\hat{u}](N,N_0) \\ &= \sum_{x_i=\xi\nu} \hat{u}_\xi(N+1)S_\nu[\hat{u}](N,N_0). \end{aligned}$$

Finally, assume the identity holds for all words up to some fixed length $n \geq 0$. Then for any $\eta \in X^n$ and $x_i \in X$ it follows that

$$\begin{aligned} &S_{x_i\eta}[\hat{u}](N+1,N_0) \\ &\quad = \hat{u}_{x_i}(N+1)S_\eta[\hat{u}](N+1,N_0) + S_{x_i\eta}[\hat{u}](N,N_0) \\ &\quad = \sum_{\eta=\xi\nu} \hat{u}_{x_i}(N+1)\hat{u}_\xi(N+1)S_\nu[\hat{u}](N,N_0) + \\ &\qquad \hat{u}_\emptyset(N+1)S_{x_i\eta}[\hat{u}](N,N_0) \\ &\quad = \sum_{x_i\eta=\xi\nu} \hat{u}_\xi(N+1)S_\nu[\hat{u}](N,N_0), \end{aligned}$$

which proves the claim for all $\eta \in X^*$. The second identity in the theorem follows directly from the first. ■

Example 3.2: Consider the case in Example 3.1 where $X = \{x_1\}$ and $\hat{u}_{x_1}(i) = \hat{u}_1(i)$ for all $i \geq N_0$. Then $c_u(i) = \sum_{k\geq 0}(\hat{u}_1(i)x_1)^k = (1-\hat{u}_1(i)x_1)^{-1}$ and

$$S[\hat{u}](N,N_0) = (1-\hat{u}_1(N)x_1)^{-1}\cdots(1-\hat{u}_1(N_0)x_1)^{-1}.$$

For example,

$$\begin{aligned} S[\hat{u}](1,0) &= S[\hat{u}](1,1)S[\hat{u}](0,0) \\ &= 1 + (\hat{u}_1(1)+\hat{u}_1(0))x_1 + (\hat{u}_1^2(1) + \\ &\quad \hat{u}_1(1)\hat{u}_1(0) + \hat{u}_1^2(1))x_1^2 + (\hat{u}_1^3(1) + \\ &\quad \hat{u}_1^2(1)\hat{u}_1(0) + \hat{u}_1(1)\hat{u}_1^2(0) + \hat{u}_1^3(0))x_1^3 + \cdots \end{aligned}$$

In this case, $S[\hat{u}](N,N_0)$ is always a rational series [1]. □

Given a generating series $c \in \mathbb{R}\langle\langle X\rangle\rangle$, the corresponding discrete-time Fliess operator is defined as

$$\hat{y}(N) = \hat{F}_c[\hat{u}](N) = \sum_{\eta\in X^*}(c,\eta)S_\eta[\hat{u}](N,N_0)$$

for any $N \geq N_0$ and $\hat{u} \in l_\infty^{m+1}(N_0)$. The series is known to always converge provided that c satisfies certain growth conditions [7]. Analogous to the continuous-time case,

$$\hat{y}(N) = \hat{F}_c[\hat{u}](N) = (c, S[\hat{u}](N,N_0)). \tag{4}$$

Consider two input sequences $(\hat{u},\hat{v}) \in l_\infty^{m+1}[N_a,N_b] \times l_\infty^{m+1}[N_c,N_d]$ with $N_b > N_a$ and $N_d > N_c$. The *catenation* of $\hat{u}$ and $\hat{v}$ at $M \in [N_a,N_b]$ is taken to be

$$\begin{aligned} &(\hat{v}\#_M\hat{u})(N) \\ &= \begin{cases} \hat{u}(N) : N_a \leq N \leq M \\ \hat{v}((N-M)+N_c) : M < N \leq M+(N_d-N_c). \end{cases} \end{aligned}$$

Define the set of sequences

$$l_{\infty,e}^{m+1}(0) := l_\infty^{m+1}(0) \cup \{\hat{\mathbf{0}}\},$$

where $\hat{\mathbf{0}}$ denotes the empty sequence with duration zero so that formally $\hat{v}\#_M\hat{\mathbf{0}} = \hat{\mathbf{0}}\#_M\hat{v} = \hat{v}$ for all $\hat{v} \in l_{\infty,e}^{m+1}(0)$. In which case, $l_{\infty,e}^{m+1}(0)$ is a monoid under this input catenation operator. Define $S[\hat{\mathbf{0}}] = 1$. The following discrete-time version of Chen's identity follows directly from Theorem 3.1.

Theorem 3.2: (Discrete-time Chen's identity) Given $(\hat{u},\hat{v}) \in l_\infty^{m+1}[N_a,N_b] \times l_\infty^{m+1}[N_c,N_d]$, $M \in [N_a,N_b]$, and $N \in [M, M+(N_d-N_c)]$ it follows that

$$S[\hat{v}]((N-M)+N_c,N_c)S[\hat{u}](M,N_a) = S[\hat{v}\#_M\hat{u}](N,N_a).$$

In particular, when $N_a = N_c = 0$ then

$$S[\hat{v}](N-M)S[\hat{u}](M) = S[\hat{v}\#_M\hat{u}](N). \tag{5}$$

Define the set of discrete-time Chen series

$$\begin{aligned} \mathcal{M}_C(X) = \{&S[\hat{u}](N) \in \mathbb{R}\langle\langle X\rangle\rangle : \hat{u} \in l_\infty^{m+1}[0,N_f], \\ &0 \leq N \leq N_f < \infty\}. \end{aligned}$$

Example 3.3: The Cauchy inverse of $c_u(i)$ in Example 3.2 is clearly $1-\hat{u}_1(i)x_1$. But there exists no obvious input $\hat{u}$ which renders this polynomial as a Chen series. Thus, the only immediate claim is that $\mathcal{M}_C(X)$ is a monoid under the Cauchy product. To recover a group structure as in the continuous-time case, the shuffle product needs to be replaced with the *quasi-shuffle* product over a larger alphabet. An analogous algebraic structure in a stochastic setting can be found in [4]. For the applications considered here, a group inverse will not be needed. □

Theorem 3.3: $\mathcal{M}_C$ is a monoid under the Cauchy product. In addition, $S : l_{\infty,e}^{m+1}(0) \to \mathcal{M}_C$ is a monoid homomorphism.

Proof: The results follow directly from (5). ■

Let $\mathrm{End}(\mathbb{R}^\infty)$ be the set of endomorphisms on the $\mathbb{R}$-vector space of real right-sided infinite sequences. This set can be viewed as the monoid of doubly infinite matrices with well defined matrix products and unit $I = \mathrm{diag}(1,1,\ldots)$. A monoid M is said to have an *infinite dimensional real representation*, Π, if the mapping $\Pi : M \to \mathrm{End}(\mathbb{R}^\infty)$ is a monoid homomorphism. The representation is *faithful* if Π is injective.

Theorem 3.4: The monoid $\mathcal{M}_C(X)$ has a faithful infinite dimensional real representation Π given by $\Pi(S[\hat{u}](N)) = \overleftarrow{\prod_{i=0}^{N}}\mathcal{S}(i)$, where $\mathcal{S}(i)$ is any matrix representation of the $\mathbb{R}$-linear map on $\mathbb{R}\langle\langle X\rangle\rangle$ given by the catenation map $\mathcal{C} : d \mapsto c_u(i)d$.

Proof: The representation claim follows from (3). To see that Π is injective, assume a fixed ordering of the words in X^*, say $\{\eta_1,\eta_2,\ldots\}$. Then define the matrix $[\mathcal{S}(i)]_{jk} = (c_u(i)\eta_k,\eta_j) = \hat{u}_\xi(i)$, where $\xi\eta_k = \eta_j$. Thus, $\mathcal{S}(i)$ is a lower triangular matrix with ones along the diagonal since $u_\emptyset(i) = 1$, $i \geq 0$. The first column is comprised of the coefficients of $c_u(i)$ in the order given to X^*. Hence, the map Π on the monoid $\mathcal{M}_C$ is injective since $c_u(i)$ can be uniquely identified from $\mathcal{S}(i) = \Pi(S[\hat{u}](i,i))$. ■

Example 3.4: Suppose $X = \{x_1\}$ as in Example 3.2. Assuming the ordering on X^* to be $\{\emptyset, x_1, x_1^2, \ldots\}$. Then for all $i \geq 0$

$$\mathcal{S}(i) = \begin{bmatrix} 1 & 0 & 0 & 0 & \cdots \\ \hat{u}_1(i) & 1 & 0 & 0 & \cdots \\ \hat{u}_1^2(i) & \hat{u}_1(i) & 1 & 0 & \cdots \\ \hat{u}_1^3(i) & \hat{u}_1^2(i) & \hat{u}_1(i) & 1 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}$$

and $c_u(i) = \sum_{k \geq 0} \hat{u}_1^k(i) x_1^k$. In addition,

$$\begin{aligned} &\Pi(S[\hat{u}](1)) \\ &\quad = \mathcal{S}(1)\mathcal{S}(0) \\ &\quad = \begin{bmatrix} 1 & 0 & 0 & 0 & \cdots \\ \hat{u}_1(1) + \hat{u}_1(0) & 1 & 0 & 0 & \cdots \\ \hat{u}_1^2(1) + \hat{u}_1(1)\hat{u}_1(0) + \hat{u}_1^2(0) & \hat{u}_1(1) + \hat{u}_1(0) & 1 & 0 & \cdots \\ \hat{u}_1^3(1) + \hat{u}_1^2(1)\hat{u}_1(0) + \hat{u}_1(1)\hat{u}_1^2(0) + \hat{u}_1^3(0) & \hat{u}_1^2(1) + \hat{u}_1(1)\hat{u}_1(0) + \hat{u}_1^2(0) & \hat{u}_1(1) + \hat{u}_1(0) & 1 & \cdots \\ \vdots & \vdots & \vdots & \vdots & \ddots \end{bmatrix}. \end{aligned}$$

Note here that the first column coincides with the coefficients of $S[\hat{u}](1)$ in Example 3.2. □

IV. Application to Time Discretization

Select some fixed $u \in L_1^m[0, T]$ with $T > 0$ finite. Choose an integer $L \geq 1$, let $\Delta := T/L$, and define the sequence of real numbers

$$\hat{u}_i(N) = \int_{N\Delta}^{(N+1)\Delta} u_i(t)\, dt, \quad i = 0, 1, \ldots, m$$

where $N \in [0, L-1]$. Note that since $u_0 = 1$, it follows that $\hat{u}_0(N) = \Delta$. The *sampling operator* Δ_S in this setting is the monoid homomorphism

$$\Delta_S : L_1^m(0) \to l_{\infty,e}^{m+1}(0) : u \mapsto \hat{u},$$

where $\Delta_S[\mathbf{0}] := \hat{\mathbf{0}}$.

Definition 4.1: The **discretization operator** $\mathcal{D} : \mathcal{G}_C(X) \to \mathcal{M}_C(X)$ is the monoid homomorphism satisfying the commutative diagram

$$\begin{array}{ccc} L_1^m(0) & \xrightarrow{\Delta_S} & l_{\infty,e}^{m+1}(0) \\ \big\downarrow{\scriptstyle P} & & \big\downarrow{\scriptstyle S} \\ \mathcal{G}_C(X) & \xrightarrow{\mathcal{D}} & \mathcal{M}_C(X) \end{array}$$

As a consequence, $\mathcal{D}$ maps any iterated integral E_η to its corresponding iterated sum S_η. Specifically, $\mathcal{D}(P[u]) = S[\hat{u}]$, where $\hat{u} = \Delta_S[u]$, and, extending $\mathcal{D}$ linearly, it follows that $\mathcal{D}(F_c[u]) = \hat{F}_c[\hat{u}]$, assuming both series converge (see [7] for details).

Example 4.1: Consider the Chen series given in Example 2.3, where $t = 2T$ and $T = 1$, that is,

$$P[v](2,1)P[u](1,0) = \mathrm{e}^{\hat{u}(1)x_2}\mathrm{e}^{\hat{u}(0)x_1}.$$

In this case

$$\begin{aligned} \mathcal{D}(\mathrm{e}^{\hat{u}(1)x_2}\mathrm{e}^{\hat{u}(0)x_1}) &= \mathcal{D}(\mathrm{e}^{\hat{u}(1)x_2})\mathcal{D}(\mathrm{e}^{\hat{u}(0)x_1}) \\ &= (1 - \hat{u}(1)x_2)^{-1}(1 - \hat{u}(0)x_1)^{-1}. \end{aligned}$$

□

V. Application to Machine Learning

Machine learning in control applications has traditionally been approached using recurrent neural networks to approximate dynamical system behavior [12], [13]. While this approach has a certain heuristic appeal, the computational costs are significant, and method is only suitable for certain classes of systems. In [8], [9] an alternative approach is given via a learning unit as shown in Figure 2. Here input-output data (u, y) from some unknown continuous-time plant (or the error system between the plant and an assumed model) is fed into the unit. The only assumption is that the data came from a system which has a Fliess operator representation $y = F_c[u]$. For example, any system modeled by a control affine analytic state space realization would fit this paradigm [11]. The generating series c for a discrete-time Fliess operator approximation $\hat{F}_c$ of the system is estimated by a parameter vector $\hat{\theta}$ using a standard mean-square error (MSE) parameter estimation algorithm with covariance resetting to enhance convergence [6, p. 65]. Control is then realized by a variety of different methods using the predicted output $\hat{y}_p = \hat{F}_c[\hat{u}]$ at time $N+1$, which can be written using (4) for $N \geq N_0$ as

$$\begin{aligned} \hat{y}_p(N+1) &= \hat{\theta}^T(N)\Pi(S[\hat{u}](N+1))e_1 \\ &= \hat{\theta}^T(N)\mathcal{S}(N+1)\Pi(S[\hat{u}](N))e_1 \end{aligned} \tag{6}$$

with $e_1 := [1\,0\,0\cdots]^T$ (see examples in [8], [9]). As only finite sums are possible in practice, all these objects are assumed to be truncated to some suitable length (for simplicity the same notation will be used). The truncation error can be estimated a priori using error bounds computed in [7]. The following simple example illustrates an implementation of this type of learning. The main advantage of the present approach is that it is easy to generalize to the multivariable setting, i.e., to an arbitrary alphabet $X = \{x_0, x_1, \ldots, x_m\}$.

Example 5.1: Consider the case where $X = \{x_1\}$ as in Examples 2.3 and 4.1. Truncating all the dimensions in (6) to length 4 gives

$$\hat{\theta}^T(N) = \begin{bmatrix} (c, \emptyset) & (c, x_1) & (c, x_1^2) & (c, x_1^3) \end{bmatrix} \tag{7a}$$

$$\mathcal{S}(N) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ \hat{u}_1(N) & 1 & 0 & 0 \\ \hat{u}_1^2(N) & \hat{u}_1(N) & 1 & 0 \\ \hat{u}_1^3(N) & \hat{u}_1^2(N) & \hat{u}_1(N) & 1 \end{bmatrix} \tag{7b}$$

$$\Pi(S[\hat{u}](N)) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ S_{x_1}(N) & 1 & 0 & 0 \\ S_{x_1^2}(N) & S_{x_1}(N) & 1 & 0 \\ S_{x_1^3}(N) & S_{x_1^2}(N) & S_{x_1}(N) & 1 \end{bmatrix}, \tag{7c}$$

where $S_{x_1^k}(N) := (S[\hat{u}](N), x_1^k)$. Therefore, the output $\hat{y}_p^3(N+1)$ is defined as the degree 3 polynomial in terms of the next applied input $\hat{u}_1(N+1)$ for $N \geq N_0 - 1$.

As a specific example, consider a plant modeled by the Fliess operator $y = F_c[u]$ with generating series $c =$

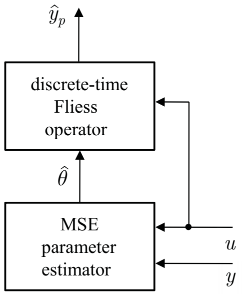

Fig. 2. Learning unit based on discrete-time Fliess operator

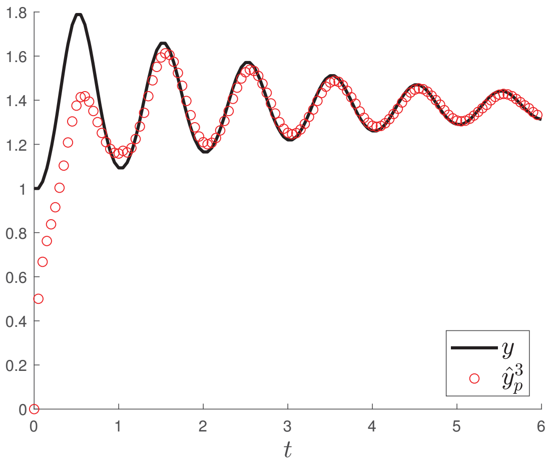

Fig. 3. Learning unit output $\hat{y}_p^3$ versus true output y in Example 5.1

$\sum_{k\geq 0} x_1^k$. The system has the state space realization

$$\dot{z}(t) = u(t), \quad z(0) = 0, \quad y(t) = \mathrm{e}^{z(t)} \tag{8}$$

since for all $t \geq 0$

$$y(t) = \sum_{k=0}^{\infty} E_{x_1}^k[u](t,0)\frac{1}{k!} = \sum_{k=0}^{\infty} E_{x_1^{\sqcup\!\sqcup\, k}\frac{1}{k!}}[u](t,0)$$
$$= \sum_{k=0}^{\infty} E_{x_1^k}[u](t,0) = F_c[u](t).$$

The output y is computed from a numerical simulation of the state space model (8) when the input $u(t) = 2e^{-t/3}\sin(2\pi t)$ is applied and plotted in Figure 3. The response of the learning unit $\hat{y}_p^3(N)$, $N \geq 0$ as implemented using (6)-(7) is also shown in the figure. The learning unit has no a priori knowledge of the system as $\hat{\theta}(-1)$ is initialized to zero. As the learning unit processes more data, its estimate of the output y improves asymptotically. □

VI. Conclusions and Future Work

After a brief overview of classical Chen series, a discrete-time notion of the concept was presented where the iterated integrals are replaced by iterated sums. In particular, it is shown that discrete-time Chen series define a monoid homomorphism between discrete-time inputs and formal power series in much the same way as in the continuous-time case. This idea also leads naturally to a discrete-time analogue of a Fliess operator. The relationship between this object and the continuous-time Fliess operator is described by another monoid homomorphism called the discretization operator. Finally, using representation theory, it is shown how to implement a known learning algorithm in the context of discrete-time Fliess operators using only matrix multiplication. Therefore, its realization is straightforward on a computational platform like MatLab.

Future work will include an implementation of the proposed learning algorithm in the full multivariable setting and an analysis of the underlying computational complexity. The long term goal is to develop an algebraic framework to analyze the interconnection of such learning units for specific control objectives.

Acknowledgments

This research was supported by the National Science Foundation under grants CMMI-1839378 and CMMI-1839387.

References

[1] J. Berstel and C. Reutenauer, *Rational Series and Their Languages*, Springer-Verlag, Berlin, 1988.

[2] K.-T. Chen, Iterated integrals and exponential homomorphisms, *Proc. Lond. Math. Soc.*, 4 (1954) 502–512.

[3] L. A. Duffaut Espinosa, K. Ebrahimi-Fard, and W. S. Gray, Combinatorial Hopf algebras for interconnected nonlinear input-output systems with a view towards discretization, in *Discrete Mechanics, Geometric Integration and Lie–Butcher Series*, K. Ebrahimi-Fard and M. Barbero Liñán, Eds., Springer Nature Switzerland AG, Cham, Switzerland, 2018, pp. 139–183.

[4] K. Ebrahimi-Fard, S. J. Malham, F. Patras, and A. Wiese, The exponential Lie series for continuous semimartingales, *Proc. R. Soc. A*, 471 (2015), no. 2184.

[5] M. Fliess, Fonctionnelles causales non linéaires et indéterminées non commutatives, *Bull. Soc. Math. France*, 109 (1981) 3–40.

[6] G. C. Goodwin and K. S. Sin, *Adaptive Filtering Prediciton and Control*, Dover Publications, Inc., Mineola, NY, 2009.

[7] W. S. Gray, L. A. Duffaut Espinosa, and K. Ebrahimi-Fard, Discrete-time approximations of Fliess operators, *Numer. Math.*, 137 (2017) 35–62.

[8] W. S. Gray, L. A. Duffaut Espinosa, and L. T. Kell, Data-driven SISO predictive control using adaptive discrete-time Fliess operator approximations, *Proc. 21st Inter. Conf. on System Theory, Control and Computing*, Sinaia, Romania, 2017, pp. 383–388.

[9] W. S. Gray, G. S. Venkatesh, and L. A. Duffaut Espinosa, Combining learning and model based control: Case study for single-input Lotka-Volterra system, *Proc. 2019 American Control Conf.*, Philadelphia, PA, 2019, submitted.

[10] W. S. Gray and Y. Wang, Fliess operators on L_p spaces: Convergence and continuity, *Systems Control Lett.*, 46 (2002) 67–74.

[11] A. Isidori, *Nonlinear Control Systems*, 3rd Ed., Springer-Verlag, London, 1995.

[12] L. Jin, P. Nikiforuk, and M. Gupta, Approximation of discrete-time state-space trajectories using dynamic recurrent neural networks, *IEEE Trans. Automat. Control*, 40 (1995) 1266–1270.

[13] C. Kambhampati, F. Garces, and K. Warwick, Approximation of non-autonomous dynamic systems by continuous time recurrent neural networks, *Proc. Inter. Joint Conf. on Neural Networks*, Como, Italy, 2000, pp. 64–69.

[14] T. J. Lyons, M. J. Caruana, and T. Lévy, *Differential Equations Driven by Rough Paths, Ecole d'Eté de Probabilités de Saint-Flour XXXIV–2004*, J. Picard, Ed., Springer, Berlin Heidelberg, 2007.

[15] C. Reutenauer, *Free Lie Algebras*, Oxford University Press, New York, 1993.

[16] H. J. Sussmann, A product expansion for the Chen series, in *Theory and Applications of Nonlinear Control Systems*, C. I. Byrnes and A. Lindquist, Eds., Elsevier Science Publishers B.V. (North Holland), New York, 1986, pp. 323–335.