2017 21st International Conference on System Theory, Control and Computing (ICSTCC)

Data-driven SISO Predictive Control Using Adaptive Discrete-time Fliess Operator Approximations

W. Steven Gray
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

Laurence T. Kell
ICCAT Secretariat
C/Corazón de María, 8
28002 Madrid, Spain

***Abstract*—Modern control theory has been applied successfully in a wide variety of engineering disciplines for decades. In sharp contrast to this situation, however, there are fields like ecology where control methodologies have not been so successful in practice. This largely due to the poor quality of models that are available and the limited amount of reliable data that can be gathered. An emerging set of control techniques known collectively as data-driven control appears to be a natural candidate for control problems in such fields. The main objective of this paper is to describe one such algorithm based on recent advances in the modeling of nonlinear input-output systems in terms of Chen-Fliess series or Fliess operators. The idea is to combine a class of discrete-time Fliess operator approximators with a standard least-squares algorithm found in adaptive control to produce a time-varying input-output model that can be used to do predictive control. As an illustration, the method is applied to a predator-prey model in order to control the population level of the prey species.**

***Keywords*—Nonlinear systems; Chen-Fliess series; predictive control; adaptive control; ecology**

I. Introduction

Modern control theory has been applied successfully in a wide variety of engineering disciplines for decades. Part of this success is due to the fact that most systems to be controlled in engineering problems can be modeled accurately using well understood physical laws. In addition, these models can be parameterized relatively easily from measurements that can be done off-line and/or repeated online. Often there is even some knowledge about the nature of the uncertainty in the system so that the controller can be robustified during the design process. In sharp contrast to this situation, however, there are fields like ecology where control methodologies have not been so successful in practice. For example, the author of the survey article [16] opens with this statement: *The goals of this study were to determine why control theory has not been used very much (in natural resource management) and why it fails...* A good example of the difficulties involved appear in fishery management, where the objective is to maximize long term catches taken by commercial fisheries while at the same time maintaining stable populations of target species such tuna and cod in the worlds oceans [2]. The constraints for the control system designer are formidable. The population dynamics are often complex and knowledge of them is incomplete. For example, it is not always clear exactly how different populations of species interact in the environment or how important processes such as density dependence operate [1]. This means that there are likely to be unmodeled dynamics present. Data for parameterizing models is limited and uncertain. Population growth rate, for example, is dependent on fertility levels, which can not be measured directly and are time-varying both seasonally and as the result of environmental disturbances such as pollution and climate change. Methods based on learning are not applicable as there is limited data available on which to train such systems. There are historical records of catches and effort obtained from fisheries, but due to misreporting there maybe significant bias in these data. Unfortunately, little else is available on which to base a plant model, especially for highly migratory species such as tuna.

An emerging set of control techniques known collectively as *data-driven* control appears to be a natural candidate for control problems in fields like ecology. A number of different definitions are available in the literature (see [14] for a survey), but what they all have in common is that the starting point for the control system designer is only input-output data, i.e., no physical model or other external knowledge is available other than perhaps that provided by practitioners who have direct experience with the system to be controlled. At the heart of this approach is how to construct a sequence of optimization problems in order to produce the desired output in the future given the input-output data up to the present. The main objective of this paper is to describe one such algorithm based on recent advances in the modeling of nonlinear input-output systems [10]. The method is based on a class of input-output models known as *Chen-Fliess series* or *Fliess operators* [5], [6]. These are generic models that can be viewed as noncommutative Taylor series in that they provide a series representation of the output in terms of iterated integrals of the input. Any analytic state space model, for example, has an input-output map that can be represented in terms of a Fliess operator. One of the main difficulties up to now in using this

978-1-5386-3842-2/17/$31.00 ©2017 IEEE

model in applications is that data is normally coming from a sampling process. While discrete-time versions of Fliess operators are available in the literature [3], [4], [18], it is not entirely clear how to employ them in data-driven control. An alternative approach, however, is described in [10], where it is shown how to use sampled-data to synthesize Fliess operator approximations with guaranteed error bounds. In this paper, the idea is to combine this class of approximators with a standard least-squares algorithm found in adaptive control [7] to produce a time-varying input-output model that can be used to do predictive control. As an illustration, the method is applied to a predator-prey model in order to control the population level of the prey species [17], [19]. This is analogous to fishery systems where the targeted species are the prey and the fishing boats are the predator. The management objectives are to maintain a stable prey population with low inter-annual variability and high productivity both in the short and long-term. The approach here is in sharp contrast to that appearing in [8], [9] where it was assumed that exact knowledge of the system is available.

The presentation is as follows. In the next section some background regarding Fliess operators and their discrete-time approximators is briefly described. In the subsequent section it is shown how to construct an adaptive scheme to identify coefficients of the generating series for an approximation. In Section IV these approximators are used to build a one-step-ahead predictive controller. The method is then demonstrated using a Lotka-Volterra population model. The conclusions of the paper and suggestions for future work are given in the final section.

II. Preliminaries

An *alphabet* $X = \{x_0, x_1, \ldots, x_m\}$ is any nonempty and finite set of noncommuting symbols referred to as *letters*. A *word* $\eta = x_{i_1} \cdots x_{i_k}$ is a finite sequence of letters from X. The number of letters in a word η, written as $|\eta|$, is called its *length*. The empty word, $\emptyset$, is taken to have length zero. The collection of all words having length k is denoted by X^k. Define $X^* = \bigcup_{k\geq 0} X^k$. This set is a monoid under the catenation product. Any mapping $c : X^* \to \mathbb{R}^\ell$ is called a *formal power series*. Often c is written as the formal sum $c = \sum_{\eta\in X^*}(c, \eta)\eta$, where the *coefficient* (c, η) is the image of $\eta \in X^*$ under c. The set of all noncommuting formal power series over the alphabet X is denoted by $\mathbb{R}^\ell\langle\langle X\rangle\rangle$. The symbol $\mathbb{R}^\ell\langle X\rangle$ represents the subset of polynomials over X. Both $\mathbb{R}^\ell\langle\langle X\rangle\rangle$ and $\mathbb{R}^\ell\langle X\rangle$ are associative $\mathbb{R}$-algebras under catenation.

A. Fliess Operators

Given a series $c \in \mathbb{R}^\ell\langle\langle X\rangle\rangle$, one can associate at least formally a causal m-input, ℓ-output operator, F_c, as follows. Let $\mathfrak{p} \geq 1$ and $t_0 < t_1$ be fixed, and assume $u : [t_0, t_1] \to \mathbb{R}^m$ is a Lebesgue measurable function. Define the norm of u to be $\|u\|_\mathfrak{p} = \max\{\|u_i\|_\mathfrak{p} : 1 \leq i \leq m\}$ with $\|u_i\|_\mathfrak{p}$ being the $L_\mathfrak{p}$-norm for each component function u_i. The set of all measurable $\mathbb{R}^m$-valued functions on the interval $[t_0, t_1]$ having a finite $\|\cdot\|_\mathfrak{p}$ norm is denoted by $L^m_\mathfrak{p}[t_0, t_1]$. A closed ball of radius $R \geq 0$ in $L^m_\mathfrak{p}[t_0, t_1]$ is defined as $B^m_\mathfrak{p}(R)[t_0, t_1] = \{u \in L^m_\mathfrak{p}[t_0, t_1] : \|u\|_\mathfrak{p} \leq R\}$. Let $C[t_0, t_1]$ be the subset of continuous functions in $L^m_1[t_0, t_1]$. For each $\eta \in X^*$ define an iterated integral E_η on $L^m_1[t_0, t_1]$ by first letting $E_\emptyset[u] = 1$ and then setting

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

where $x_i \in X$, $\bar{\eta} \in X^*$, and $u_0 := 1$. The *Fliess operator* corresponding to the generating series c is then the weighted sum of iterated integrals

$$y(t) = F_c[u](t) = \sum_{\eta\in X^*} (c, \eta)\, E_\eta[u](t, t_0). \qquad (1)$$

To establish convergence of this series, assume there exist real numbers $K_c, M_c > 0$ so that

$$|(c, \eta)| \leq K_c M_c^{|\eta|}|\eta|!, \quad \forall\eta \in X^*. \qquad (2)$$

It is shown in [12] that under such circumstances the series (1) converges uniformly and absolutely so that F_c describes a well defined mapping from $B^m_\mathfrak{p}(R)[t_0,\, t_0+T]$ into $B^\ell_\mathfrak{q}(S)[t_0,\, t_0+T]$ for some $S > 0$ provided $\bar{R} := \max\{R, T\} < 1/M_c(m+1)$, and $\mathfrak{p}, \mathfrak{q} \in [1, \infty]$ satisfy $1/\mathfrak{p} + 1/\mathfrak{q} = 1$. (Define $|z| := \max_i |z_i|$ whenever $z \in \mathbb{R}^\ell$.) The operator F_c in this context is said to be *locally convergent*, and the collection of all generating series c satisfying the growth condition (2) is denoted by $\mathbb{R}^\ell_{LC}\langle\langle X\rangle\rangle$. When c satisfies the more restrictive (slower) growth condition

$$|(c, \eta)| \leq K_c M_c^{|\eta|}, \quad \forall\eta \in X^*, \qquad (3)$$

the series (1) also establishes an input-output operator from the L_p extended space $L^m_{\mathfrak{p},e}(t_0)$ into $C[t_0, \infty)$, where

$$L^m_{\mathfrak{p},e}(t_0) := \{u : [t_0, \infty) \to \mathbb{R}^m : u_{[t_0,t_1]} \in L^m_\mathfrak{p}[t_0, t_1], \\ \forall t_1 \in (t_0, \infty)\},$$

and $u_{[t_0,t_1]}$ is taken as the restriction of the input u to the interval $[t_0, t_1]$ [12]. The operator F_c in this case is called *globally convergent*, and the set of all generating series with a growth bound (3) is denoted by $\mathbb{R}^\ell_{GC}\langle\langle X\rangle\rangle$.

A Fliess operator F_c is called *realizable* on a ball $B^m_\mathfrak{p}(R)[t_0, t_0 + T]$ whenever there exists a system of n ordinary differential equations and a set of ℓ output functions,

$$\dot{z}(t) = g_0(z(t)) + \sum_{i=1}^{m} g_i(z(t))\, u_i(t), \quad z(t_0) = z_0 \qquad (4a)$$

$$y_j(t) = h_j(z(t)), \quad j = 1, \ldots, \ell, \qquad (4b)$$

where each analytic vector field g_i is written in terms of local coordinates on a neighborhood $\mathcal{W}$ of z_0, every real-valued function h_j is analytic on $\mathcal{W}$, and (4a) has a solution z well defined on $[t_0, t_0 + T]$ for any input $u \in B^m_\mathfrak{p}(R)[t_0, t_0 + T]$ satisfying $y_j(t) = F_{c_j}[u](t) = h_j(z(t))$, $j = 1, \ldots, \ell$. It is easily verified that for any $\eta = x_{i_k} \cdots x_{i_1} \in X^*$

$$(c_j, \eta) = L_{g_\eta} h_j(z_0) := L_{g_{i_1}} \cdots L_{g_{i_k}} h_j(z_0), \qquad (5)$$

where $L_{g_i} h_j$ denotes the *Lie derivative* of the output h_j with respect to the vector field g_i. A given operator F_c can be shown to be realizable if and only if its generating series $c \in \mathbb{R}^\ell_{LC}\langle\langle X\rangle\rangle$ has finite Lie rank [5], [6], [15].

B. Discrete-time Approximations of Fliess Operators

The goal of this section is to define the set of discrete-time approximators to be employed by the control algorithm developed in subsequent sections. See [10] for additional details. Inputs in this case will be real-valued sequences from the space

$$l_\infty^{m+1}[N_0] := \{\hat{u} = (\hat{u}(N_0), \hat{u}(N_0+1), \dots) : \\ |\hat{u}(N)| < \hat{R}_{\hat{u}} < \infty, \forall N \geq N_0\},$$

where $\hat{u} := [\hat{u}_0, \hat{u}_1, \dots, \hat{u}_m]^T$ and defining $|\hat{u}(N)| = \max_{i\in\{0,1,\dots,m\}} |\hat{u}_i(N)|$. Therefore, the norm $\|\hat{u}\|_\infty := \sup_{N\geq N_0} |\hat{u}(N)|$ is finite for every admissible $\hat{u}$.

Definition 2.1: Given a generating series $c \in \mathbb{R}^\ell\langle\langle X\rangle\rangle$, the corresponding **discrete-time Fliess operator** is defined as

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

for any $N \geq 1$, where

$$S_{x_i\eta}[\hat{u}](N) = \sum_{k=1}^{N} \hat{u}_i(k) S_\eta[\hat{u}](k), \tag{6}$$

when $x_i \in X$, $\eta \in X^*$, and $\hat{u} \in l_\infty^{m+1}[1]$. By assumption, $S_\emptyset[\hat{u}](N) := 1$.

Following [13], 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

$$\hat{u}_i(N) = \int_{(N-1)\Delta}^{N\Delta} u_i(t)\, dt, \quad i = 0, 1, \dots, m \tag{7}$$

where $N \in [1, L]$. Note that since $u_0 = 1$ it follows that $\hat{u}_0(N) = \Delta$. A truncated version of $\hat{F}_c$ will be useful,

$$\hat{y}(N) = \hat{F}_c^J[\hat{u}](N) := \sum_{j=0}^{J} \sum_{\eta\in X^j} (c,\eta) S_\eta[\hat{u}](N), \tag{8}$$

since numerically only finite sums can be computed. The main assertion is that the class of truncated, discrete-time Fliess operators acts as a set of universal approximators for their continuous-time counterparts. The following two theorems from [10] provide the corresponding error bounds for the two different coefficient growth rates (2) and (3), respectively.

Theorem 2.1: Let $c \in \mathbb{R}_{LC}^\ell\langle\langle X\rangle\rangle$ with growth constants $K_c, M_c > 0$. If $u \in B_1^m(R)[0,T]$ with $\bar{R} := \max\{R,T\} < 1/M_c(m+1)$ then for any fixed integer $J \geq 0$

$$\left|F_c[u](T) - \hat{F}_c^J[\hat{u}](L)\right| \\ \leq \hat{e}(J) + e(J) + K_c \frac{1-\hat{s}^{J+1}}{1-\hat{s}} O\left(\frac{1}{L^2}\right)$$

as $L \to \infty$, where

$$\hat{e}(J) = \frac{K_c}{L}\left[\frac{\hat{s}^2}{(1-\hat{s})^3} - \frac{J(J+1)\hat{s}^{(J+1)}}{2(1-\hat{s})} - \frac{J\hat{s}^{(J+2)}}{(1-\hat{s})^2} - \frac{\hat{s}^{J+2}}{(1-\hat{s})^3}\right],$$

$$e(J) = K_c \frac{s^{J+1}}{1-s},$$

$\hat{s} = M_c(m+1)L\|\hat{u}\|_\infty$, $s = M_c(m+1)\bar{R}$, and $\hat{u}$ is as defined in (7).

Theorem 2.2: Let $c \in \mathbb{R}_{GC}^\ell\langle\langle X\rangle\rangle$ with growth constants $K_c, M_c > 0$. If $u \in B_1^m(R)[0,T]$ then for any fixed integer $J \geq 0$

$$\left|F_c[u](T) - \hat{F}_c^J[\hat{u}](L)\right| \\ \leq \hat{e}(J) + e(J) + K_c\, \mathrm{e}^{\hat{s}}\Gamma(J+1,\hat{s})\, O\left(\frac{1}{L^2}\right)$$

as $L \to \infty$, where

$$\hat{e}(J) = K_c\, \mathrm{e}^{\hat{s}}\Gamma(J+1,\hat{s})\frac{\hat{s}^2}{2L}, \quad e(J) = K_c\, \mathrm{e}^{s}(1-\Gamma(J+1,s)),$$

$\hat{s} = M_c(m+1)L\|\hat{u}\|_\infty$, $s = M_c(m+1)\bar{R}$, $\bar{R} = \max\{R,T\}$, and $\hat{u}$ is as defined in (7).

III. Adaptive Discrete-Time Fliess Operator Approximations

The main objective of this section is to introduce adaptation for the truncated discrete-time Fliess operator coefficients via a standard least-squares algorithm [7]. For simplicity, only the SISO case is considered. If the operator is truncated to words of length J then (8) can be written in the form

$$\hat{y}(N) = \phi^T(N)\theta_0, \quad N \geq 1, \tag{9}$$

where

$$\phi(N) = [S_{\eta_1}[\hat{u}](N)\ S_{\eta_2}[\hat{u}](N) \cdots S_{\eta_l}[\hat{u}](N)]^T \tag{10a}$$

$$\theta_0 = [(c,\eta_1)\ (c,\eta_2) \cdots (c,\eta_l)]^T \tag{10b}$$

with $l = 2^{J+1} - 1$ and assuming some ordering $(\eta_1, \eta_2, \dots)$ has been imposed on the words in X^*. Here lexicographical ordering was chosen assuming $x_0 < x_1$. Note, in particular, that in this instance $\hat{y}(N)$ depends on the input sequence up to time N rather than $N-1$, which is more common. If some estimate of θ_0 is available at time $N-1$, say $\hat{\theta}(N-1)$, then (9) gives a corresponding estimate of $\hat{y}(N)$:

$$\hat{y}_p(N) := \phi^T(N)\hat{\theta}(N-1). \tag{11}$$

The following least-squares algorithm can be used to update coefficients:

$$\hat{\theta}(N) = \hat{\theta}(N-1) + g(N-1)e(N) \tag{12a}$$

$$e(N) = y(N\Delta) - \phi^T(N)\hat{\theta}(N-1) \tag{12b}$$

$$g(N-1) = \frac{P(N-2)\phi(N)}{1+\phi^T(N)P(N-2)\phi(N)} \tag{12c}$$

$$P(N-1) = P(N-2) - \frac{P(N-2)\phi(N)\phi^T(N)P(N-2)}{1+\phi^T(N)P(N-2)\phi(N)} \tag{12d}$$

for any $N \geq 1$ with the initial estimate $\hat{\theta}(0)$ given, and $P(-1)$ is any positive definite matrix P_0. Setting $P_0 = I$, it is known that this algorithm minimizes the performance index

$$\mathbf{J}_{\bar{N}}(\theta) := \sum_{N=1}^{\bar{N}} [y(N\Delta) - \phi^T(N)\theta]^2 + \frac{1}{2}\|\theta - \hat{\theta}(0)\|^2.$$

with respect to the parameter θ. It should be stated that since the model class consists of truncated versions of $\hat{F}_c$, there is no reason to expect the parameter vector $\hat{\theta}(N)$ to converge to c in any fashion as N increases. But this is not a problem since the only objective is to ensure that the underlying input-output map F_c is well approximated by $\hat{F}^J_{\hat{\theta}(N)}$. On the other hand, the theory presented in the previous section guarantees that if the underlying system has a Fliess operator representation then the class of discrete-time approximators provides a feasible set of limit points for the sequence $\hat{\theta}(N)$, $N \geq 0$.

Now in order to get a fully real-time adaptive algorithm, a difference equation is needed to update the data vector $\phi(N)$ as defined in (10a). This result is presented next. For convenience define $\hat{u}_\xi(N) = \hat{u}_{i_k}(N) \cdots \hat{u}_{i_1}(N)$ for any $\xi = x_{i_k} \cdots x_{i_1} \in X^*$ and $N \geq 1$. Let $\hat{u}_\emptyset(N) := 1$. The following lemma states that the update equation for the iterated sums defined in (6) is computed in terms of the Cauchy product of the formal power series $c_u(N+1) := \sum_{\eta \in X^*} \hat{u}_\eta(N+1)\eta$ and $S[\hat{u}](N) := \sum_{\eta \in X^*} S_\eta[\hat{u}](N)\eta$.

Lemma 3.1: For any $\eta \in X^*$ and $N \geq 1$

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

Therefore,

$$\phi(N+1) = \begin{bmatrix} \hat{u}_\emptyset(N+1)S_\emptyset[\hat{u}](N) \\ \hat{u}_\emptyset(N+1)S_{x_0}[\hat{u}](N) + \hat{u}_{x_0}(N+1)S_\emptyset[\hat{u}](N) \\ \hat{u}_\emptyset(N+1)S_{x_1}[\hat{u}](N) + \hat{u}_{x_1}(N+1)S_\emptyset[\hat{u}](N) \\ \sum_{x_0^2=\xi\nu} \hat{u}_\xi(N+1)S_\nu[\hat{u}](N) \\ \sum_{x_0x_1=\xi\nu} \hat{u}_\xi(N+1)S_\nu[\hat{u}](N) \\ \sum_{x_1x_0=\xi\nu} \hat{u}_\xi(N+1)S_\nu[\hat{u}](N) \\ \sum_{x_1^2=\xi\nu} \hat{u}_\xi(N+1)S_\nu[\hat{u}](N) \\ \vdots \\ \sum_{\eta_\ell=\xi\nu} \hat{u}_\xi(N+1)S_\nu[\hat{u}](N) \end{bmatrix}. \quad (13)$$

Proof: The proof is by induction on the length of η. When $\eta = \emptyset$ then trivially $S_\emptyset[\hat{u}](N+1) = 1 = \hat{u}_\emptyset(N+1)S_\emptyset[\hat{u}](N)$ as claimed. If $\eta = x_i \in X$ then from (6)

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

Finally, assume the proposition 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

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

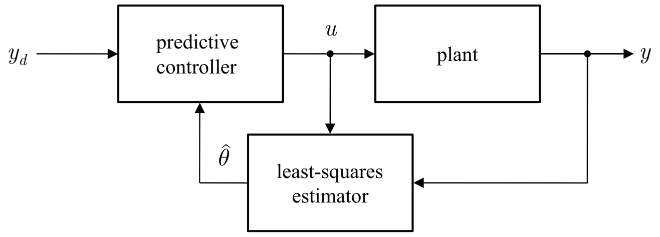

Fig. 1. Closed-loop system with predictive controller

which proves the claim for all $\eta \in X^*$. ■

In summary then, combining (11), (12) and (13), one has a fully inductive algorithm which can predict $\hat{y}(N+1)$ given the next input $\hat{u}(N+1)$:

$$\begin{aligned} \hat{y}_p(N+1) &= \phi^T(N+1)\hat{\theta}(N) \\ &= \sum_{i=1}^{\ell} \left[\sum_{\eta_i=\xi\nu} \hat{u}_\xi(N+1)S_\nu[\hat{u}](N) \right] \hat{\theta}_i(N) \\ &= \sum_{i=1}^{\ell} \sum_{\eta_i=\xi\nu} \Delta^{|\xi|_{x_0}} \hat{u}^{|\xi|_{x_1}}(N+1)S_\nu[\hat{u}](N)\hat{\theta}_i(N) \\ &= \mathcal{U}^T(N+1)\mathcal{S}(N)\hat{\theta}(N) \\ &=: \mathcal{Q}(\hat{u}(N+1)), \end{aligned}$$

where

$$\mathcal{U}^T(N+1) := [1 \ \ \hat{u}(N+1) \ \ \hat{u}^2(N+1) \cdots \hat{u}^J(N+1)]$$

and $\mathcal{S}(N) \in \mathbb{R}^{J+1 \times \ell}$ with entries depending solely on the sums $\{S_\eta[\hat{u}](N) : |\eta| \leq J\}$. So $\mathcal{Q}(\hat{u}(N+1))$ is a polynomial in $\hat{u}(N+1)$ of at most degree J. This motivates the predictive control scheme described in the next section.

IV. Predictive Control

If y_d is a desired output known to be in the range of F_c (see [11]) then a suitable input u can be approximated by a piecewise constant function taking values for $N \in [1, L]$ equivalent to

$$\hat{u}(N) := \arg\min_{\hat{u}(N)} \|\hat{y}_d(N) - \mathcal{Q}(\hat{u}(N))\|^2, \quad (14)$$

where $\hat{y}_d(N) = y_d(N\Delta)$. This control law can be implemented practically using the MatLab command `fminbnd`, which finds the minimum of a function over a given bounded interval of $\mathbb{R}$. The corresponding closed-loop system is shown in Figure 1. The method is illustrated by the following example.

The population dynamics of n species in competition is often described by the classical Lotka-Volterra model

$$\dot{z}_i = \beta_i z_i + \sum_{j=1}^{n} \alpha_{ij} z_i z_j, \ \ i = 1, \ldots, n, \quad (15)$$

where z_i is the biomass of the i-th species, β_i represents the growth rate of the i-th species, and the parameter α_{ij} describes the influence of the j-th species on the i-th species (normally, $\alpha_{ii} = 0$). This model can exhibit a wide range of behaviors under various conditions including the presence of multiple stable equilibria, stable limit cycles, and even chaotic behavior [17], [19]. Environmental factors and human interactions can also affect the dynamics of animal populations. This is modeled mathematically by allowing the parameters appearing in (15) to be time dependent, specifically $\beta_i(t)$ and $\alpha_{ij}(t)$ are assumed to be integrable functions of time. A subset of these parameters can be viewed as inputs u_i, $i = 1, \ldots, m$, if they can be actuated. It is often impractical to estimate the state $z = [z_1 \cdots z_n]^T$ directly, so output functions

$$y_j = h_j(z), \quad j = 1, \ldots, \ell \tag{16}$$

are used to model measurement processes. As the inputs enter the dynamics linearly, it is clear that (15)-(16) can be written in the form (4). In which case, the input-output system $u \mapsto y$ must have an underlying Fliess operator representation F_c with generating series c given by (5). But as discussed in the introduction, precise knowledge of the g_i and the initial condition $z(0)$ is usually not available in practice. In which case, one option is to use the adaptive method described in the previous section to synthesize a sequence of truncated, discrete-time Fliess operator approximators using only input-output data, and then employing one-step-ahead predictive control via (14). Of course any control system designed using actual knowledge of the system is likely to outperform such a predictive controller. In addition, the desired trajectory y_d has to be selected using some physical insight into the system, otherwise the control problem may not be well-posed and yield exceedingly poor closed-loop performance.

Consider the special case of a predator-prey system

$$\dot{z}_1 = \beta_1 z_1 - \alpha_{12} z_1 z_2$$
$$\dot{z}_2 = -\beta_2 z_2 + \alpha_{21} z_1 z_2,$$

where z_1 is assumed to be the prey species, and (15) has been re-parameterized so that $\beta_i, \alpha_{ij} > 0$. This system has precisely two equilibria when all the parameters are fixed, namely, a stable equilibrium at the origin and a saddle point at $z_e = (\beta_2/\alpha_{21}, \beta_1/\alpha_{12})$. The vector fields in the first quadrant are complete thereby giving concentric periodic trajectories about z_e. Setting $y = z_1$, there are only four possible SISO maps depending on which parameter function is considered as the input. For example, if it is assumed that $u(t) = \beta_1(t)$ then the resulting system takes the form

$$\begin{bmatrix} \dot{z}_1(t) \\ \dot{z}_2(t) \end{bmatrix} = \begin{bmatrix} -\alpha_{12} z_1(t) z_2(t) \\ -\beta_2 z_2(t) + \alpha_{21} z_1(t) z_2(t) \end{bmatrix} + \begin{bmatrix} z_1(t) \\ 0 \end{bmatrix} u(t)$$
$$y(t) = \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} z_1(t) \\ z_2(t) \end{bmatrix}$$

with initial conditions $z_1(0) = z_{1,0}$ and $z_2(0) = z_{2,0}$. For simplicity, assume all the constant system parameters are set to unity. Initializing the model at $[z_{1,0}, z_{2,0}]^T = [1/6, 1/6]^T$ and applying the constant input $u(t) = \beta_1(t) = 1$ yields the first orbit shown in black in Figure 2. Suppose the biological goal is to reduce the large population fluctuations to avoid the very low prey population levels appearing during certain portions of this orbit. A new final orbit can be identified by setting $\beta_1 = 1.5$ and $[z_{1,0}, z_{2,0}]^T = [1/2, 1/2]^T$ to produce the red orbit in Figure 2. The control objective is to synthesize a control u to achieve this orbit transfer in some smooth fashion. The desired orbit transfer is the dashed blue line shown in Figure 2. It was designed simply by scaling the $y_1 = z_1$ population as the system moved along the initial orbit until the transfer trajectory intercepted the final orbit. Since only SISO control is being employed here, there is little direct control over the z_2 population. In which case, the exact entry point into the final orbit can not be arbitrarily selected. In addition, once the final orbit is achieved, additional control will be needed to maintain this orbit, but that portion of the problem is not addressed here. Instead, the focus is on synthesizing an input for the orbit transfer problem.

TABLE I
PARAMETERS FOR LOTKA-VOLTERRA PREDICTIVE CONTROLLER: DISCRETE-TIME FLIESS OPERATOR APPROXIMATOR(TOP), LEAST-SQUARES PARAMETER ESTIMATOR (BOTTOM)

L	J	T	Δ
100	3	6	0.06

$\hat{\theta}(0)$	$P(0)$	$\lvert\hat{u}(N)\rvert$ bound
0	1000*triu(toeplitz(flip(1:15)))	2

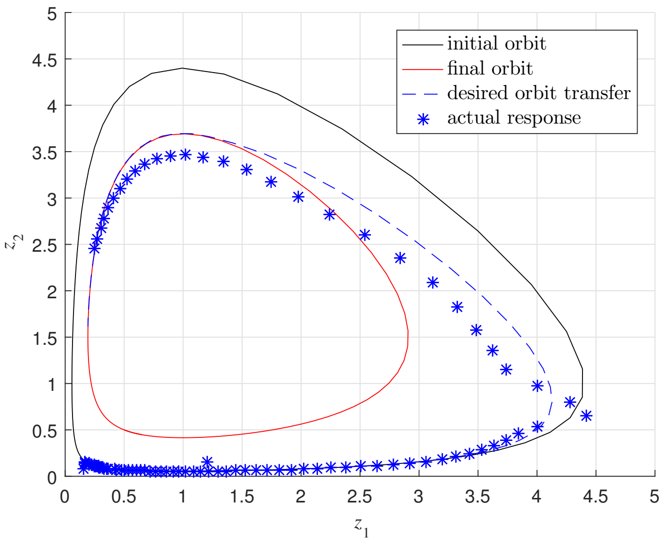

Fig. 2. Orbits of the Lotka-Volterra system

A MatLab simulation of the closed-loop system in Figure 1 was performed, where the plant is the Lotka-Volterra system under consideration. The controller parameters used are summarized in Table I. The initial condition for the parameter $\hat{\theta}(0)$ was set to zero. So the control system has no a priori knowledge of the plant. The resulting orbit transfer is shown in Figure 2 and compared against the desired trajectory. The final orbit is achieved at $t = 5.4$. The corresponding time trajectories for the populations and the applied input are shown in Figures 3 and 4, respectively. The initial spike in the input

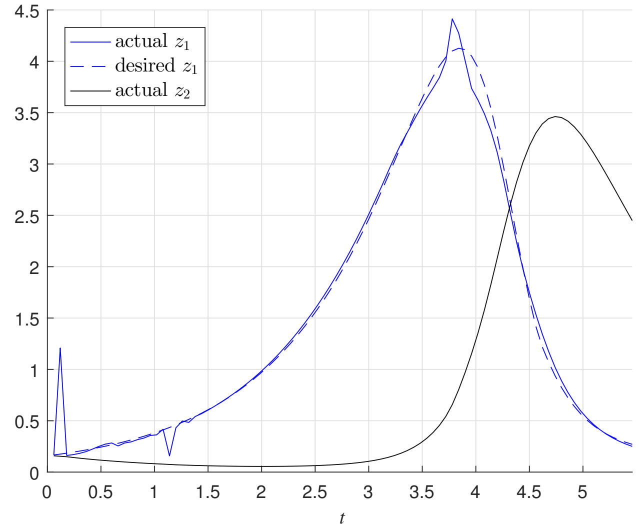

Fig. 3. Population trajectories during the orbit transfer

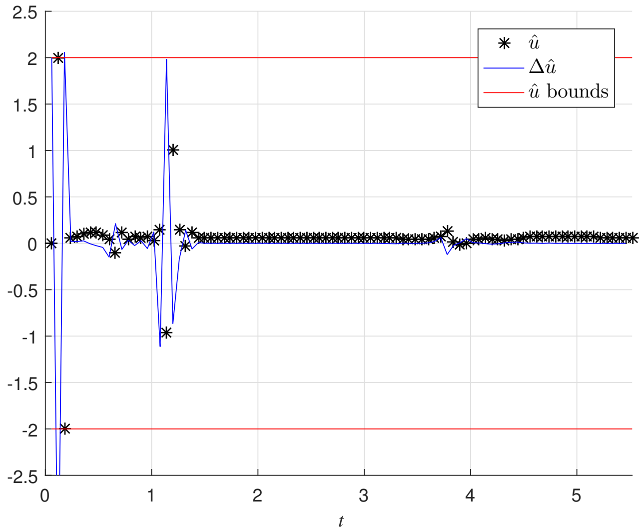

Fig. 4. Applied input $\hat{u}$, change in input $\Delta\hat{u}(N) := \hat{u}(N) - \hat{u}(N-1)$, and input bound during the orbit transfer

just after $t = 0$ is the system adjusting to the nonzero initial conditions of the plant. The two smaller spikes at $t = 1$ and $t = 3.8$ are the result of numerical instabilities that are inherent when optimization problems are solved near singular points, in this case, changes in the degree of the polynomial $\mathcal{Q}$. The bound set on $\hat{u}$ within the optimization routine `fminbnd` generally helps contain these instabilities. If this bound is set too high, and the controller is run long enough, the closed loop system can become unstable. Fortunately, these numerical instabilities are usually easy to detect and indicate that some tuning of the parameters is needed. Decreasing L often helps simply by reducing the number of optimizations problems that must be solved per unit time. Another common approach to addressing such problems in predictive control in general is to expand the time horizon beyond one-step-ahead prediction. But this becomes numerically expensive, especially with the nonlinear method being used here. The question of whether this can be done efficiently will be explored in future work.

V. Conclusions and Future Work

A SISO data-driven predictive controller is described based on discrete-time Fliess operator approximators whose coefficients were identified from only input-output data using a standard least-squares algorithm found in adaptive control. As an illustration, the method is applied to a predator-prey model in order to reduce the fluctuations in the population level of the prey species. Future work will include an extension of the algorithm to the multivariable case, exploring the use of longer time horizon predictors to reduce numerical sensitivity, and an investigation of how measurement noise affects closed-loop performance. Higher fidelity populations models will also be utilized.

References

[1] P. M. Cury, J.-M. Fromentin, S. Figuet, and S. Bonhommeau, Resolving Hjort's dilemma: How is recruitment related to spawning stock biomass in marine fish?, *Oceanography*, 27 (2014) 42–47.

[2] C. T. T. Edwards and D. J. Dankel, *Management Science in Fisheries - An Introduction to Simulation-based Methods*, Routledge, Milton, Oxfordshire, UK, 2016.

[3] M. Fliess, Un codage non commutatif pour certains systèmes échantillonnés non linéaires, *Inf. Control*, 38 (1978) 264–287.

[4] M. Fliess, Generating series for discrete-time nonlinear systems, *IEEE Trans. Automat. Control*, AC-25 (1980) 984–985.

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

[6] M. Fliess, Réalisation locale des systèmes non linéaires, algèbres de Lie filtrées transitives et séries génératrices non commutatives, *Invent. Math.* **71** (1983) 521–537.

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

[8] W. S. Gray, L. A. Duffaut Espinosa, and K. Ebrahimi-Fard, Analytic left inversion of SISO Lotka-Volterra models, *Proc. 49th Conf. on Information Sciences and Systems*, Baltimore, Maryland, 2015, pp. 1–6.

[9] W. S. Gray, L. A. Duffaut Espinosa, and K. Ebrahimi-Fard, Analytic left inversion of multivariable Lotka-Volterra models, *Proc. 54th IEEE Conf. on Decision and Control*, Osaka, Japan, 2015, pp. 6472–6477.

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

[11] W. S. Gray, L. A. Duffaut Espinosa, and M. Thitsa, Left inversion of analytic nonlinear SISO systems via formal power series methods, *Automatica*, 50 (2014) 2381–2388.

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

[13] L. Grüne and P. E. Kloeden, Higher order numerical schemes for affinely controlled nonlinear systems, *Numer. Math.*, 89 (2001) 669–690.

[14] Z.-S. Hou and Z. Wang, From model-based control to data-driven control: Survey, classification and perspective, *Inform. Sci.* 235 (2013) 3–35.

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

[16] C. Loehle, Control theory and the management of ecosystems, *J. Appl. Ecol.*, 45 (2006) 957–966.

[17] R. M. May and W. J. Leonard, Nonlinear aspects of competition between three species, *SIAM J. Appl. Math.*, 29 (1975) 243–253.

[18] S. Monaco and D. Normand-Cyrot, Functional expansions for nonlinear discrete-time systems, *Math. Syst. Theory*, 21 (1989) 235–254.

[19] S. Smale, On the differential equations of species in competition, *J. Math. Biol.*, 3 (1976) 5–7.