Neurocomputing 000 (2018) 1–8

Contents lists available at ScienceDirect

Neurocomputing

journal homepage: www.elsevier.com/locate/neucom

Neural inverse optimal control for discrete-time impulsive systems

Gustavo Hernandez-Mejia [a], Alma Y. Alanis [b], Esteban A. Hernandez-Vargas [a,*]

[a] Frankfurt Institute for Advanced Studies, Ruth-Moufang-Strasse 1, Frankfurt am Main 60438, Germany
[b] CUCEI, Universidad de Guadalajara, Guadalajara C.P.44430, Mexico

ARTICLE INFO

Article history:
Received 2 February 2018
Revised 30 April 2018
Accepted 3 June 2018
Available online xxx

Communicated by Dr. K Chan

Keywords:
Neural networks
Recurrent high order neural networks
Inverse optimal control
Impulsive control
Viral infection treatment

ABSTRACT

Impulsive systems describe processes with at least one state variable is impulsively changeable. The design of optimal control policies in impulsive systems is a complex task. In order to relax the solution for the Hamilton-Jacobi-Bellman equation, a meaningful cost functional can be proposed a posteriori in the inverse optimal problem. The main contribution of this paper is a neural inverse optimal control for discrete-time impulsive systems. Control policies for discrete-time impulsive systems are derived by combining inverse optimal control into a recurrent high order neural network (RHONN) trained with the Extended Kalman filter (EKF). The neural network avoids the development of a mathematical model to represent the studied system. For illustration, we apply the proposed neurocontrol to personalized drug treatment in influenza infection disease, whose nonlinear model is included and described for completeness. The robustness of the proposed framework is tested through Monte Carlo simulations.

1. Introduction

Recurrent neural networks are a well-established methodology that allows solving difficult problems such as identification and control of complex systems [1]. As a result, the use of recurrent neural networks (RNNs) for modeling and learning has rapidly increased in recent years [1–3]. These networks allow the identification of dynamical systems in form of high dimensional nonlinear state space models [1]. In addition, new training algorithms such as those based on the extended Kalman filter have been raising due to different properties. Some of these features are important to improve technical issues related to local minima, slow learning rate, high sensitivity to initial conditions, among others [4,5]. Therefore, the Extended Kalman filter (EKF) is an important learning tool to train neural networks [4].

Recurrent high order neural networks (RHONNs) schemes present many convenient features for modeling and control of nonlinear systems [6,7]. For instance, these networks trained with the EKF allow to reduce the epoch size and the number of required neurons [4,8]. Discrete-time RHONNs present more interactions among the neurons, the network design is flexible and allow the incorporation of previous information about the system structure into the RHONN model [9,10]. In addition, the discrete-time neural networks are better fitted for real-time implementations [11]. These features favor forming a discrete-time representation of dynamical systems [10].

In recent years, there exists a trend towards discrete-time control rather than analog control of dynamic systems. This tendency is mainly due to the advantages of working with digital instead of continuous-time signals [9]. A controller based on a plant model may not perform as desired because of uncertain parameters or unmodeled dynamics as well as internal and external disturbances [8]. A way to solve these issues is on the basis of a black/gray box approach such as RHONNs, which allow identifying the dynamics of the plant to be controlled. Once the weights of the network are adapted, the RHONN model dynamics are similar to the real system dynamics, even in presence of disturbances. As a result, a controller based on a RHONN model may increase its robustness [10].

Diverse control techniques have been developed in the last years *e.g.* sliding-mode control [12], robust control [13], optimal control [14], inverse optimal control [15,16], and impulsive control [17]. However, the optimal control can determine the input that will force a process to satisfy physical constraints while a performance criterion is minimized [16]. To this end, it is required to solve the associated Hamilton–Jacobi–Bellman (HJB) equation, which is not an easy task [9]. An alternative way to solve directly the HJB equation is based on the inverse optimal control, which allows to develop a stabilizing control law that optimizes a cost functional [9,15,16].

Impulsive control refers to systems that at least one state variable is impulsively changeable leading to impulsive differential equations [17]. The impulsive control has many applications such

* Corresponding author.
E-mail address: vargas@fias.uni-frankfurt.de (E.A. Hernandez-Vargas).

https://doi.org/10.1016/j.neucom.2018.06.034

as HIV treatment [18–20], influenza treatment [21], vaccination strategies at a population level [22], economics and biological systems [23].

In this work, we establish a neural inverse optimal impulsive control framework for discrete-time systems. The controlled system is identified by a RHONN, which is then used in the control design process. In principle, the neurocontroller does not require previous knowledge of the original system to be controlled. The scheme herein presented is applied to the problem of influenza virus infection treatment. The control policies are based on the discrete-time RHONN to forecast drug amounts for within-host influenza virus infection. The main goal is to reach similar efficacies with respect to the current Food and Drug Administration (FDA) medication but reducing the amount of the administered drug.

2. Neural identification

Notation. The notation employed through this paper is as follows: $\Re$ denotes the set of real numbers. $\Re^n$ denotes the set of $n \times 1$ column vectors. $\Re^{n\times m}$ denotes the set of $n \times m$ real matrices. $\aleph \in \mathbb{Z}^+ \cup 0 = \{0, 1, 2, \ldots\}$ denotes the set of non-negative integers. P^n denotes the set of $n \times n$ positive-definite matrices. N^n is a $n \times n$ nonnegative definite matrix and $S \subset \aleph$ is the resetting set, which is reserved for control instants action. Moreover, $(\cdot)^T$ denotes the matrix transpose and $(\cdot)^{-1}$ stands for the matrix inverse.

2.1. Recurrent high order neural networks

In this subsection, we study the identification problem of a general nonlinear system. Consider the following discrete-time system

$$x(k+1) = F(x(k), u(k)) + \varepsilon_{zi}, \tag{1}$$

where $x(k) \in \Re^n$ is the state vector of the system. $u(k) \in \Re^m$ is the control input. F: $\Re^n \times \Re^m \to \Re^n$ is a nonlinear function and $k \in \aleph$ is the sampling step. In addition, to identify the discrete-time nonlinear system (1) we can employ the following series-parallel structure discrete-time RHONN [6,24]

$$\chi_i(k+1) = \omega_i^T z_i(x(k), u(k)), \tag{2}$$

where $i = 1, \ldots, n$. n is the state dimension. $\chi_i(k)$ is the state of the ith neuron and ω_i is the respective on-line adapted weight vector. $z_i(x(k), u(k))$ is given by

$$z_i(x(k), u(k)) = \begin{pmatrix} z_{i_1} \\ z_{i_2} \\ \vdots \\ z_{i_{L_i}} \end{pmatrix} = \begin{pmatrix} \prod_{j\in I_1} \xi_{i_j}^{d_{i_j}(1)} \\ \prod_{j\in I_2} \xi_{i_j}^{d_{i_j}(2)} \\ \vdots \\ \prod_{j\in I_{L_i}} \xi_{i_j}^{d_{i_j}(L_i)} \end{pmatrix}, \tag{3}$$

$$\xi_i = \begin{pmatrix} \xi_{i_1} \\ \vdots \\ \xi_{i_n} \\ \xi_{i_{n+1}} \\ \vdots \\ \xi_{i_{n+m}} \end{pmatrix} = \begin{pmatrix} \varphi(x_1) \\ \vdots \\ \varphi(x_n) \\ u_1 \\ \vdots \\ u_m \end{pmatrix}, \tag{4}$$

$$\varphi(v) = \frac{1}{1+e^{-av}}, \quad a > 0. \tag{5}$$

L_i is the respective number of high order connections. m is the number of external inputs. $\{I_1, I_2, \ldots, I_{L_i}\}$ is a collection of non-ordered subsets of $\{1, \ldots, n+m\}$, with $d_{ij}(k)$ being a nonnegative integer. $u = \{u_1, u_2, \ldots u_m\}^T$ in (4) is the input vector to the neural network with $\varphi(\cdot)$ defined by (5) where v can be a real value variable [4,6,24]. To identify the general discrete-time nonlinear system (1) using the RHONN (2), consider that ε_{zi} is a bounded approximation error that can be reduced by increasing the number of the adjustable weights [6].

Assumption 1. There exist an ideal weight vector ω_i^* such that $\|\in_{zi}\|$ can be minimized on a compact set $\Omega_{z_i} \subset \Re^{L_i}$. The ideal weight vector ω_i^* is an artificial quantity for the analysis approach. It is assumed that this vector exists and is constant but unknown [6].

2.2. Network training with the Extended Kalman filter

The training of a neural network is a process in which the neural network learns a specific task, this training can be on-line or off-line [25,26]. The EKF-based training algorithm estimates the neural network weights which become the state of the network. The error between the measured output of the dynamical system and the output of the neural network is considered as additive white noise [9,26]. The training of RHONN (2) is performed on-line by the modified EKF-based algorithm, as follows [4]:

$$\omega_i(k+1) = \omega_i(k) + \eta_i K_i(k) e_i(k), \tag{6}$$

$$K_i(k) = \rho_i(k) H_i(k) M_i(k), \tag{7}$$

$$\rho_i(k+1) = \rho_i(k) - K_i(k) H_i^T(k) \rho_i(k) + Q_i(k), \tag{8}$$

with

$$M_i(k) = [\varrho_i(k) + H_i^T(k)\rho_i(k)H_i(k)]^{-1},$$
$$e_i(k) = x_i(k) - \chi_i(k), \tag{9}$$

$$H_{ij} = \left[\frac{\partial \chi_i(k)}{\partial \omega_{ij}(k)}\right]^T, \tag{10}$$

where $i = 1, \ldots, n$ and $j = 1, \ldots, L_i$. $e_i(k) \in \Re$ is the respective identification error. $\rho_i(k) \in \Re^{L_i \times L_i}$ is the prediction error associated covariance matrix at the step k. $\omega_i \in \Re^{L_i}$ is the weight vector considered as state of the network. $\chi_i(k)$ is the ith neural network state. $x_i(k)$ is the ith plant state, n is the number of states. $K_i \in \Re^{L_i}$ is the Kalman gain vector. $Q_i \in \Re^{L_i \times L_i}$ is the state noise associated covariance matrix. $\varrho_i \in \Re$ is the measurement noise associated covariance. $H_i \in \Re^{L_i}$ is a vector, in which each entry H_{ij} is the derivative of one of the neural network state $\chi_i(k)$, with respect to one neural network weight ω_{ij} defined in (10). A common practice is that ρ_i and Q_i are initialized as diagonal matrices, with entries $\rho_i(0)$ and $Q_i(0)$, respectively [4]. It is important to remark that for the EKF, $H_i(k)$, $\rho_i(k)$ and $K_i(k)$ are bounded [27]. The stability of a RHONN trained with the EKF to identify a discrete-time nonlinear system has been previously studied [24].

3. Inverse optimal impulsive neurocontrol

In this section, we present inverse optimal control policies for a discrete-time impulsive system. First, we set the optimal control framework and the dynamical system to be analyzed and controlled in terms of the RHONN identified states. Then, an inverse optimal control approach is established.

3.1. Impulsive optimal control methods

This subsection briefly discusses the optimal control methodology and its properties. Consider the discrete-time nonlinear dynamical system of the form

$$x(k+1) = f(x(k)) + gu(k), \tag{11}$$

Please cite this article as: G. Hernandez-Mejia et al., Neural inverse optimal control for discrete-time impulsive systems, Neurocomputing (2018), https://doi.org/10.1016/j.neucom.2018.06.034