IJE TRANSACTIONS A: Basics Vol. 29, No. 4, (April 2016) 490-499

International Journal of Engineering

Journal Homepage: www.ije.ir

Designing a Neuro-Sliding Mode Controller for Networked Control Systems with Packet Dropout

M. H. Vali, B. Rezaie*, Z. Rahmani

Facultyof Electrical and Computer Engineering, Babol Noshirvani University of Technology, Babol, Iran

PAPER INFO

Paper history:
Received 18 February 2016
Received in revised form 29 March 2016
Accepted 14 April 2016

Keywords:
Networked Control Systems
Packet Dropouts
Sliding Mode Control
Genetic Algorithm
Radial-basis Function Neural Network

ABSTRACT

This paper addresses control design in networked control system by considering stochastic packet dropouts in the forward path of the control loop. The packet dropouts are modelled by mutually independent stochastic variables satisfying Bernoulli binary distribution. A sliding mode controller is utilized to overcome the adverse influences of stochastic packet dropouts in networked control systems. Firstly, to determine the parameters of switching function used in the sliding mode control design, an improved genetic algorithm is applied. The proposed improved genetic algorithm provides a fast convergence rate and a proper dynamic performance in comparison with conventional genetic algorithms especially in online control applications. Then, an adaptive neural sliding mode control based on radial-basis function neural network approximation is proposed to eliminate chattering phenomenon in the sliding mode control. A numerical example is given to illustrate the effectiveness of the proposed controller in networked control systems. The results show that the proposed controller provides high-performance dynamic characteristics and robustness against plant parameter variations and external disturbances.

***doi**: 10.5829/idosi.ije.2016.29.04a.07*

NOMENCLATURE

A, B, C	Constant matrices in continuous-time model	P_c, P_m	Crossover rate and mutation rate probability
b_i, c_i, h_i	Parameter of neural network	s	Sliding surface
C^T	Parameter vector of sliding surface	u, u_c	Control inputs received to actuator andproduced by controller
d	External pertubation and unmodeled dynamics in continuous-time model	u_{eq}, u_s	Equivalent and discontinuous control inputs
$\tilde{d}$	External pertubation and unmodeled dynamics in discrete-time model	w_i	Weights of neural network
$\hat{d}$	Upper bound of d in continuous-time model	x, y	State and output vectors of the system
d_l, d_u	Lower and upper bound of d in discrete-time model	**Greek Symbols**	
e	Error	β_k	Mutually independent Bernoulli binary distributed white sequences
f, g	Nonlinear vector functions	δ_d	DSMC parameter
f_i, f_{ave}, M, R_i	Parameter of genetic algorithm		SMC design parameter
MSE	Mean square error	λ_i	Parameter of sliding surface
$P\{\}$	Probability function	$\Phi_\tau, \Gamma_\tau, D_\tau$	Constant matrices in discrete-time model

*Corresponding Author's Email: brezaie@nit.ac.ir (B. Rezaie)

Please cite this article as: M. H. Vali, B. Rezaie, Z. Rahmani, Designinga Neuro-Sliding Mode Controller for Networked Control Systems with Packet Dropout, International Journal of Engineering (IJE), TRANSACTIONS A: Basics Vol. 29, No. 4, (April 2016) 490-499

1. INTRODUCTION

Networked Control System (NCS) is a kind of distributed control systems wherein the feedback control loops are closed via communication network.

Traditional control systems use a point-to-point structure in which the plants, sensors and actuators are directly wired to various controllers. A point-to-point architecture does not address the reliability and flexibility that are required for modern control systems. Moreover, it cannot support the industrial requirements for smart sensors and distributed control due to its high cost of implementation.

NCS provides several advantages such as flexible architectures, high reliability, less wiring and easy maintenance compared with the conventional point-to-point control systems. However, the insertion of the communication network will lead to time delay and data packet dropout. The existence of time delays and dropped packets complicate the analysis and design of NCSs and may cause instability and poor performance.

Because of the inherent complexity of NCS with taking into account network-induced delays and/or packet dropouts, new challenges have appeared in modeling, analysis and design of NCS.

Network packet dropouts happen in NCSs, when there are node failures or message collisions. Packet dropout is an important aspect in the analysis and synthesis of NCSs and this issue has received wide attention recently. In the literature, the stochastic models of packet dropouts in NCSs are categorized into two types: Markov chains [1] and Bernoulli binary distributed white sequence [2]. In [3-5], the stabilization of NCSs with packet dropout has been investigated. Also, a few results have been presented for control design issue. In [6], a fuzzy logic speed controller is proposed to NCS with time delays and packet dropouts. The fuzzy tracking control is studied in [7] for a class of nonlinear networked control systems with a prescribed H_∞ tracking performance. In addition, H_∞ controller design has been presented for NCS with time delay and packet dropout.

Although the prior studies provide many advantages, there are usually some conservative limitations in the application of these methods, and they cannot effectively overcome the adverse effects of packet dropouts. Due to the complexity of NCS, simple and efficient methods are required to overcome such difficulties. Sliding mode control (SMC) is a robust and efficient method for controlling nonlinear systems in presence of disturbance amd uncertainties. To the best of our knowledge, sliding mode controllers have not been used for controlling NCSs with packet dropout.

This paper addresses the design of SMC for NCSs with packet dropout in forward path. The controller-to-actuator packet dropout is modelled by Bernoulli binary distribution.

In recent years, evolutionary optimization techniques have been usually applied to determine such sliding surface parameters in SMC method in order to improve the system performance [8-10].

In this paper a sliding mode controller is applied to overcome the adverse influences of stochastic packet dropout in NCSs. To determine the proper parameters of SMC, an improved genetic algorithm (GA) is applied. Then, an adaptive neural SMC (ANSMC) based on radial-basis function (RBF) neural network approximation is proposed to eliminate chattering phenomenon in SMC.

The main contributions of the paper can be summarized as follows:

- Utilizing SMC into NCS using a novel control structure
- Applying an improved GA to adjust the SMC parameters
- Using RBF neural network with adaptive learning algorithm as approximator in order to reduce the chattering in SMC

Because of the simultaneous use of neural network and improved SMC in this control scheme, ANSMC uses the great advantages of both methods. Therefore, the proposed method can effectively control the systems with high nonlinearity and also in presence of serious uncertainty in networked structure and external disturbances.

The paper is organized as follows. In Section 2, formulation of data packet dropout in forward path and model of this packet dropouts by Bernoulli binary distribution are presented. In Section 3, the basic concepts of design for SMC and also discrete-time SMC (DSMC) will be described by considering data packet dropout in NCS. Section 4 gives a brief review on the genetic algorithms and will introduce an improved GA. In Section 5, an adaptive neural SMC based on RBF neural network approximation is proposed to eliminate chattering phenomenon in SMC. Simulation results are shown in Section 6. Finally, Section 7 concludes the paper.

2. PROBLEM FORMULATION

The structure of NCS with random data packet dropouts in forward path is shown in Figure 1. In this figure $u(k) \in R^m$ is the control input received to actuator and $u_c(k) \in R^m$ is the current controller produced by controller without packet dropouts.

The plant is assumed to be of the following form:

$$
\begin{aligned}
x(k+1) &= Ax(k) + Bu(k) \\
y &= Cx(k)
\end{aligned}
\tag{1}
$$

where $x(k) \in R^n$ is the state vector, $A \in R^{n\times n}$, $B \in R^{n\times m}$ and $C \in R^{p\times n}$ are known constant matrices.

In the feedback path, there are messages such as sensor measurements and calculated control signals. It is assumed that the actuator has a buffer to hold most recent data as a newly received measurement when a packet dropout happens. When the controller output $u_c(k)$ is lost in the networks, the actuator will hold the control input at the previous value, and therefore $u(k)=u(k-1)$; whereas when the controller output $u_c(k)$ is successfully transmitted to the actuator, we have $u(k)=u_c(k)$.

Assuming the probability of a packet dropout in the network as Bernoulli random binary distribution, the control input can be described as:

$$u(k)=\beta_k u_c(k)+(1-\beta_k)u(k-1) \tag{2}$$

where the stochastic variable β_k is the mutually independent Bernoulli binary distributed white sequences with values of 0 and 1 that satisfying Equations (3) and (4) :

$$P\{\beta_k=1\}=\beta \tag{3}$$

$$P\{\beta_k=0\}=1-\beta \tag{4}$$

where P{ β_k } means the occurrence probability of the event β_k . From Equations (1), (2), (3), and (4), the NCS model can be described as:

$$\begin{aligned} x(k+1)&=Ax(k)+B\beta_k u_c(k)+B(1-\beta_k)u(k-1) \\ y&=Cx(k) \end{aligned} \tag{5}$$

The proposed methodology in this paper is considered to apply to the systems which can be described by Equation (1). However, due to applicability of SMC and neural networks in a general class of nonlinear systems and it can considerably enhance the results for nonlinear systems. Moreover, due to the capability of ANSMC against uncertainties and external disturbances, it can be used for the systems in the networked environment in which the uncertainty in the form of packet dropouts exist as described in (5).

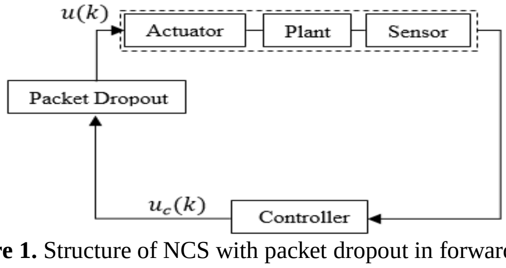

Figure 1. Structure of NCS with packet dropout in forward path

We also consider that a bounded external disturbance can be added to the first row of (5).

3. SLIDING MODE CONTROL

Control in the presence of uncertainty and external disturbances is one of the main topics of modern control theory**.** Robust control approaches are used to overcomesuch model uncertainties and external disturbances. SMC as an important robust control methods provides a systematic strategy for maintaining stability and desired performance in the presence of external disturbances and model uncertainties.

The principle of the SMC is to force the trajectories of the system to converge towards a pre-defined surface and keep them on the close neighbourhood of this sliding surface in spite of the existence of uncertainties and the disturbances.

SMC design includes two parts. The first part involves the design of a sliding surface and the second one concerns with the selection of a control law to drive the variables to zero in finite time.

3. 1. Continuous-time Sliding Mode Control We consider a general non-linear continuous system represented by Equation (6):

$$\dot{x}=f(x)+g(x)u+d(x) \tag{6}$$

where $x=[x_1,x_2,\ldots,x_n]^T$ is the state vector, u is control law, *f(x)* and *g(x)* are two nonlinear vector functions and *d(x)*denotes the external perturbation and unmodeled dynamics. *d(x)* is limited by a known vector function $\hat{d}$ as expressed in Equation (7):

$$\|d(x)\|\le \hat{d} \tag{7}$$

The surface is also defined by a set of relations between the state variables of the system according to the control objectives.

A typical form of the sliding surface is as below:

$$s(x)=c^T x=\sum_{i=1}^{n}\lambda_i x_i(t)=\lambda_1 x_1(t)+\ldots+\lambda_n x_n(t) \tag{8}$$

where $c=[\lambda_1,\lambda_2,\ldots,\lambda_n]^T$ is a vector of parameters in which λ_i is constant.

The function $s(x)$ should be selected in such a way that it vanishes, i.e., $s(x)=0$ [11, 12].

The control $u=u(x_1,x_2,\ldots,x_n)$ drives the state variables to the sliding surface in finite time and keeps them on the surface thereafter in the presence of the bounded disturbance.

The control law is composed of two parts as below:

$$u = u_{eq} + u_s \tag{9}$$

where u_{eq} is the equivalent control and u_s is the discontinuous control.

The equivalent control is only effective once the states trajectory are on the surface. Thus, the equivalent control that maintains the sliding mode is the input u_{eq} satisfying Equation (10):

$$\dot{s} = c^T f(x) + c^T g(x) u_{eq}(t) + c^T d(x) = 0 \tag{10}$$

Assuming $c^T g(x) \neq 0$, the equivalent control may be calculated as below:

$$u_{eq}(t) = -\left(c^T g(x)\right)^{-1} c^T f(x) \tag{11}$$

The discontinuous control provides robustness against the high frequency unmodelled dynamics and can be represented by Equation (12):

$$u_s = -\varepsilon \left(c^T g(x)\right)^{-1} sign(s) \tag{12}$$

where $\varepsilon > c^T \hat{d}$ is a design parameter [13, 14].
A control input can now be constructed using Equations (9), (11), and (12), for the system (6) in the form of Equation (13):

$$u(t) = -\left(c^T g(x)\right)^{-1} \left(c^T f(x) + \varepsilon \operatorname{sgn}(s)\right) \tag{13}$$

3. 2. Discrete-time Sliding Mode Control In the recent years, researches have been carried out in the field of DSMC. DSMC is the discrete-time counterpart of the continuous-time SMC discussed in the previous section.

In the case of DSMC, measurement and control are performed within time intervals and the control signal is constant during these time intervals [15]. According to [16], we have:

$$s(k+1) - s(k) = -\varepsilon \tau sgn\left(s(k)\right) \tag{14}$$

where τ is the sampling interval.
For a LTI system:

$$x(k+1) = \Phi_\tau x(k) + \Gamma_\tau u(k) + D_\tau \tilde{d}(k) \tag{15}$$

where Φ_τ, Γ_τ and D_τ are matrices with appropriate dimension and $\tilde{d}(k)$ is the disturbance vector representing the effects of unmodeled dynamics and external disturbances. Since, it is assumed that $\tilde{d}(k)$ is bounded, it is reasonable to suppose that $d(k) = c^T D_\tau \tilde{d}(k)$ will also be bounded. Let the bounds be:

$$d_l \leq d(k) \leq d_u \tag{16}$$

$$d_0 = \frac{d_l + d_u}{2} \tag{17}$$

$$\delta_d = \frac{d_u - d_l}{2} \tag{18}$$

where d_l and d_u are upper and lower bound of *d(x)*.
According to [16]:

$$\begin{aligned} s(k+1) - s(k) = {} & -\varepsilon \tau sign\left(s(k)\right) + c^T D_\tau \tilde{d}(k) - d_0 \\ & -\delta_d sign\left(s(k)\right) \end{aligned} \tag{19}$$

Considering stable sliding surface, $s(k) = c^T \mathrm{x}(\mathrm{k})$, the control law, can be written as:

$$\begin{aligned} u(k) = {} & -\left(c^T \Gamma_\tau\right)^{-1} \left(c^T \Phi_\tau - c^T\right) x(k) \\ & -\left(c^T \Gamma_\tau\right)^{-1} \left(d_0 + (\delta_d + \varepsilon \tau) sign\left(s(k)\right)\right) \end{aligned} \tag{20}$$

where $\varepsilon > d_u$ is control parameter.

The important task is to optimize the control law in Equation (20). Hence, it is necessary to choose the proper parameters of the sliding surface. If the parameters of *c* are properly chosen, then the control system performance could be greatly improved using online techniques.

In this paper, a SMC approach is investigated in which the suitable parameters of SMC are determined using an improved GA is applied.

4. GENETIC ALGORITHM

GA is an optimization technique proposed by John Holland. GA simulates the phenomena that takes place in the evolution of species based on the concepts of evolution theory.

GA is a useful approach in the cases that an effective and efficient searching is required. GA works with a set of artificial creature (string) called population. Each population consist of some individual or chromosome. In every generation, GA generates a set of offsprings from old population according to a fitness function. The fitness function represents the performance, i.e., the higher fitness value, the better performance of the system. Each individual (chromosome) consists of a data structure, and represents a possible solution in the search space of the problem.

GA evolves into new generations of individuals by using knowledge from previous generations. One or more individuals after a series of evolution are the optimal solutions.

GA starts with an initial population of individuals, which is usually generated randomly. By exchanging the information between every individual, GA keeps the better schemata, which may yield higher fitness, from generation to generation such that the performance can be improved.

Population size is the key factor in GA optimization process. If this size is too small, GA may converge too fast, but GA may reject important information contained in the other individuals and lead to premature convergence. If the size is too large, the search speed is too slow and the operation will waste time, which is difficult to use for on line applications.

New generations are created according to emulate biological evolution by means of genetic operators such as reproduction, crossover and mutation etc.

Selecting some fine individuals from current population to pass down the next generation according to fitness value of each individual is named as selection. Crossover composes a pair with every two individuals of the population and exchanges parts of the chromosomes by some probability (P_c). Changing one or several gene values by some probability (P_m) for each individual of the population is called mutation. New individuals are generated in the population by mutation.Crossover and mutation produce new chromosomes and are used to keep the variety of the population and to make genetic algorithms keeping active random search ability. Therefore, it is helpful for escape to converge the local optimal solution.

4. 1. Fast Genetic Algorithm In simple genetic algorithm , crossover and mutation are applied equally to all chromosomes in spite of their fitness. Thus, the simple genetic algorithm usually cannot converge to a global optimal solution. In the recent years, many researchers have tried to improve the performance of GA. This paper presents a fast genetic algorithm (FGA) [17] which includes a lot of improvements in population, selection, crossover and mutation in comparison with simple GA.

In FGA optimization, the larger selection probability of individuals will product more offspring. They come into the next generation with no crossover and mutation operations, otherwise it will be cut. It is shown as:

$$R_i = M \times \frac{f_i}{f_{avg}} = MP_i \tag{21}$$

where R_i is the individual amount in the next generation. M is the population size. Moreover, f_i and f_{avg} are current fitness values and average fitness values.

In the conventional mechanism, the crossover and mutation rates are held constant. So, we use an adaptive method in which crossover rate and mutation rate probability will diminish with generation increase, i.e.:

$$P_c = P_{cl} - P_{cl} \times \frac{i}{I} \tag{22}$$

$$P_m = P_{ml} - P_{ml} \times \frac{i}{I} \tag{23}$$

where i is the number of current iterations and I the total iterations. It is assumed that P_{cl} and P_{ml} are 0.6 and 0.001, respectively.

In this study, FGA is used to find the proper parameters of SMC λ_1 , λ_2. The block diagram of the proposed scheme is depicted in Figure 2.

In this figure, $u(k) \in R^m$ is the control input and $u_c(k) \in R^m$ is the current controller output without packet dropouts.

5. ARTIFICIAL NEURAL NETWORK

Using sign-function in SMC control law yields a finite amplitude and finite frequency zigzag motion in the sliding surface. This effect is called chattering which is undesirable in practice, since it involves high control activity and may excite high frequency dynamics neglected in the modeling.Chattering can also affect the system's steady-state accuracy and even make the system unstable. Chattering must be eliminated or at least reduced in the system's response by the aid of the controller.

In order to solve the chattering phenomenon, a usual method is that the discontinuous "sign" term is replaced by a continuous smooth approximation "sat" and "tanh".

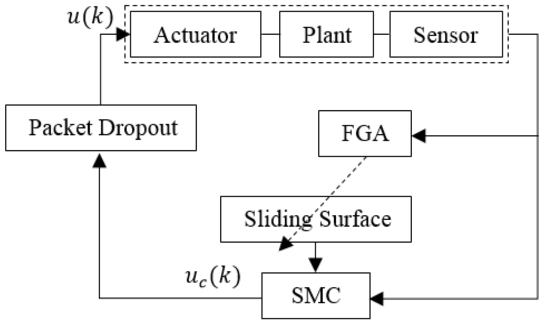

Figure 2. NCS with SMC based on FGA

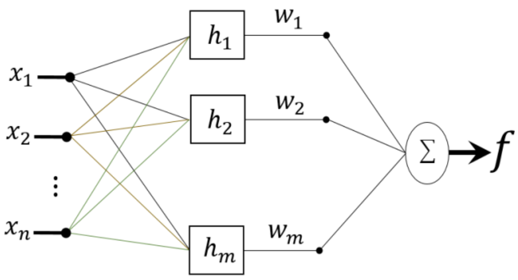

Figure 3. Structure of RBF neural network

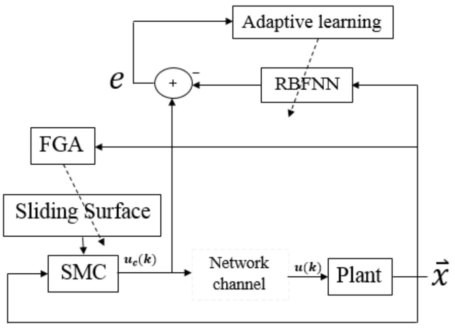

Figure 4. NCS with ANSMC controller

This approach can reduce chattering and control effort, but reduces the system's steady-state accuracy. One of best choices to eliminate the chattering phenomena is the use of artificial neural network (ANN).

ANN not only possesses the ability of parallel computing and self-learning, but also it has an enormous potentiality in dealing with high nonlinearity and serious uncertainty. However, there still exist some defects such as weak stability, slow response, etc. Combining neural network with SMC not only has strong robustness which can resist system perturbation and external disturbance, but also is able to eliminate the chattering phenomena. Therefore, we propose a novel strategy which combine RBF neural network and SMC.

RBF neural network is a three-layer feedforward networks which provides a nonlinear mapping from input layer to output layer, while the mapping from hidden layer to the output layer is linear. The structure of RBF network is shown in Figure 3.

In RBF network, $X = [x_1 \quad x_2 \quad \dots \quad x_n]^T$ is the input vector, i indicates the input layer of the network, j represents the hidden layer nodes in the network, h_j is defined as the following Gaussian transfer function:

$$h_j = exp\left(-\frac{X - c_j{}^2}{2b_j^2}\right) \quad j = 1, 2, \dots, m \tag{24}$$

where $c_j = [c_{j1} \quad c_{j2} \quad \dots \quad c_{jn}]^T$ is the central vector of j-th node. $\boldsymbol{B} = [b_1 \quad b_2 \quad \dots \quad b_m]^T$ is the base width vector of the neural network and $b_j > 0$ is the base width constant of j-th node. In addition, $W = [w_1 \quad w_2 \quad \dots \quad w_m]^T$ is the weight vector.

The output of the neural network is as below [18]:

$$y_i = \left(\sum_{j=1}^{m} w_j h_j \left(X - c_j\right)\right) + b_i \tag{25}$$

The error between outputs of the neural network and the controller can be defined as:

$$e_i = \frac{1}{2}\left[y_i(k) - y_m(k)\right]^2 \tag{26}$$

Based on the gradient descent algorithm, the iterative algorithm for updating the output weights and the central vector can be written as:

$$w_j(k+1) = w_j(k) - \eta \frac{\partial MSE}{\partial w_j}|_{w_j} \tag{27}$$

$$c_{ji}(k+1) = c_{ji}(k) - \eta \frac{\partial MSE}{\partial c_{ji}}|_{c_{ji}} \tag{29}$$

where, η is the learning rate and MSE is defined as:

$$MSE = \frac{1}{N}\sum_{i=1}^{N} e_i^2 \tag{30}$$

For modifying state responses in SMC based on FGA, an ANSMC based on RBF neural network approximation, is proposed. The proposed scheme can provide good robustness and dynamic performance in the presence of disturbance. Also, to eliminate the chattering phenomenon in SMC, we can use this strategy. Figure 4 shows structure of ANSMC based on RBF neural network approximation.

6. SIMULATION RESULTS

To illustrate the merits of the proposed method, we apply the method to a simple LTI discrete-time system described by:

$$x(k+1) = \begin{bmatrix} 0 & 1 \\ -1 & 1 \end{bmatrix} x(k) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(k) + \begin{bmatrix} 0 \\ 0.01 \end{bmatrix} \sin(0.1k) \tag{31}$$

where $x = [x_1, x_2]^T$ is the state vector, u is the control law and $sin(0.1k)$denotes the external perturbation and unmodeled dynamics.

According to Equations (16)-(18), d_0, d_u, d_l and δ_d can be obtained as $d_l = 0$, $d_u = 0.02$ and $d_0 = \delta_d = 0.01$ where d_l and d_u are upper and lower bounds of $d(k)$. According to Equation (5), the NCS model for system (31) can be described as:

$$\begin{aligned} x(k+1) &= \begin{bmatrix} 0 & 1 \\ -1 & 1 \end{bmatrix} x(k) + \beta_k \begin{bmatrix} 0 \\ 1 \end{bmatrix} u_c(k) + (1-\beta_k) \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(k-1) \\ &+ \begin{bmatrix} 0 \\ 0.01 \end{bmatrix} \sin(0.1k) \end{aligned} \tag{32}$$

where $x = [x_1, x_2]^T$ is the state vector, $u(k)$ is the control input and $u_c(k)$, the current controller output without packet dropouts. β_k is the mutually independent Bernoulli binary distributed white sequences with values of 0 and 1. Sliding function can be defined as:

$$s(k)=\lambda_1 x_1(k)+\lambda_2 x_2(k)=c^T x(k) \tag{33}$$

where $c=[\lambda_1,\lambda_2]^T$ is the parameter vector in which λ_i is constant. The control signal is defined as:

$$u_c(k)=-\left(c^T\begin{bmatrix}0\\1\end{bmatrix}\right)^{-1}\left(c^T\begin{bmatrix}0 & 1\\-1 & 1\end{bmatrix}-c^T\right)x(k)$$
$$-\left(c^T\begin{bmatrix}0\\1\end{bmatrix}\right)^{-1}\left(0.01+(0.01+\eta\tau)sign(s(k))\right) \tag{34}$$

where $\eta=0.2$ is the controller parameter and $\tau=0.1$ is the sampling time.

Figure 5 and Figure 6 illustrates the performance of conventional SMC when it is directly applied to the plant and when it is applied in NCS structure, respectively. Sliding function is designed as $s(x)=[-0.8\ \ 1]x(k)$ and initial states value is chosen as $[0\quad 0]^T$ for both cases and packet dropout rate (β) is 0.8 for the case of NCS. Comparing Figures 5 and 6, it can be seen that packet dropout has adverse influence on NCS and leads to poor performance in state responses. Therefore, utilizing SMC by itself cannot resolve the problem and advanced methods have to be used to overcome the effects of packet dropouts in NCS.

Applying the control scheme of Figure 2, the results of designing simple SMC based on FGA for NCS are depicted in Figure 7.

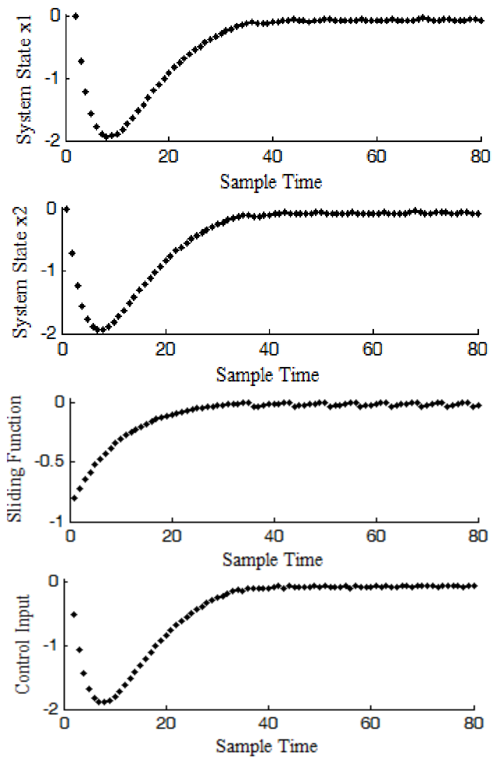

Figure 5. States of system, sliding function and control input in conventional SMC with $s(x)=[-0.8\ \ 1]x(k)$

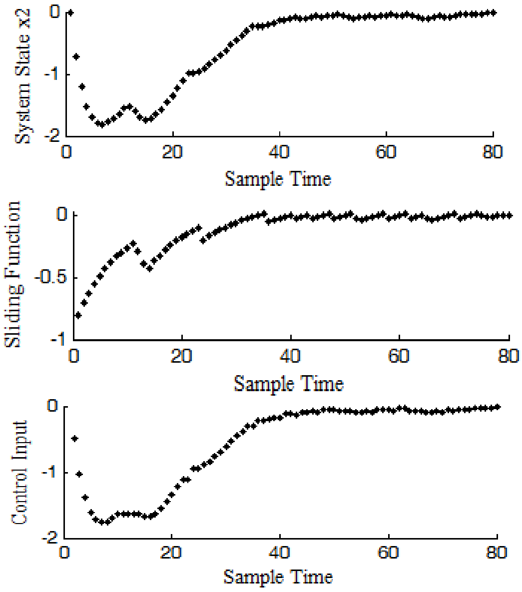

Figure 6. States of system, sliding function and control input in conventional SMC used in NCS with $\beta=0.8$ and $s(x)=[-0.8\ \ 1]x(k)$

The population size, maximum iteration, initial rates of crossover and mutation are considered as 30, 30, 0.8 and 0.05, respectively. Vector c is the vector of the selective parameters (design variables) of the SMC that are determined by FGA. Initial states value is chosen as $[0\quad 0]^T$ and packet dropout rate (β) is 0.8.

Comparing Figures 6 and 7, it can be concluded that improved parameters of switching function enhance the performance of SMC and overcomes the adverse influences of stochastic packet dropouts in NCSs.

CPU time of simulations for GA and FGA for NCS with $\beta=0.8$ is 22.75s and 12.57s, respectively. Thus, low CPU time is a merit of using FGA than GA and makes FGA as a proper choice for online control applications.

Figure 8 illustrates the performance of ANSMC in NCS. Initial states value is chosen as $[0\quad 0]^T$ and packet dropout rate (β) is 0.8. $X=[x_1\quad x_2]^T$ that obtained from FGA is inputs of RBF neural. Number of hidden layer nodes (neurons) are 4. For each Gaussian function, the parameters of c_j are designed as 3.0. The weight vector and central vector can be obtained by Equations (27) and (28).

Comparing Figures 7 and 8, it can be concluded that ANSMC causes good performance and can effectively eliminate the chattering phenomenon and can also overcome the adverse influences of stochastic packet dropouts in NCSs comparing with the simple SMC. In order to make a qualitative comparison among the applied methods, i.e., SMC without NCS, SMC in

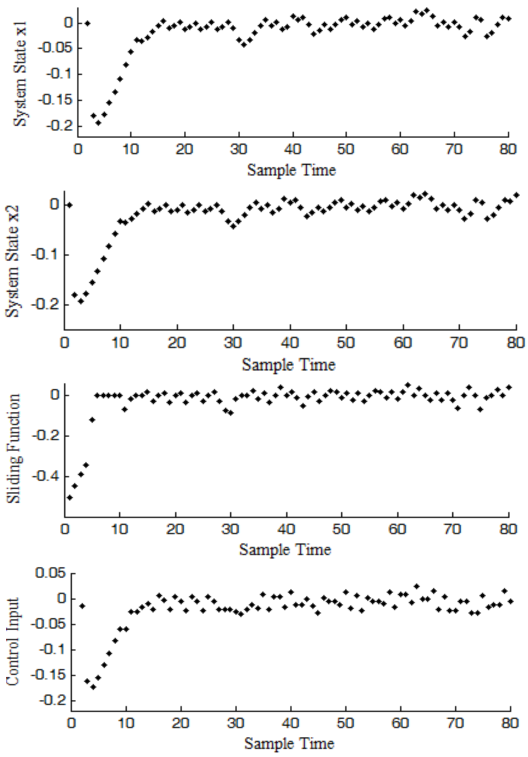

Figure 7. states of system, sliding function and control input in NCS with $\beta = 0.8$ by silding mode controller based FGA

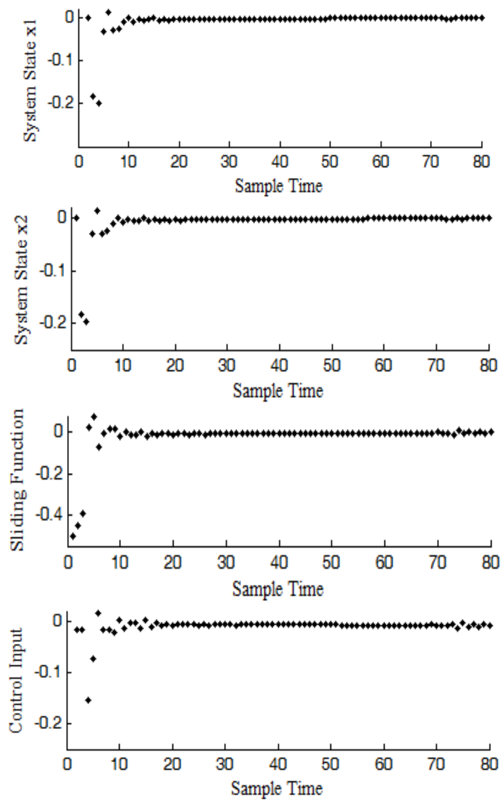

Figure 8. States of system in NCS with $\beta = 0.8$ by neuro sliding mode controller based on FGA

TABLE 1. Comparison of the applied methods

	$\frac{1}{N}\sum e^2$	$\sum e^2$	**Maximum $\lvert u \rvert$**	**Average (e)**	**Reaching Time**
A	0.0067	0.5362	1.9731	-0.4388	36
B	0.0046	0.3668	-1.7829	-0.3891	39
C	0.00015	0.0115	0.1970	-0.0224	14
D	0.00012	0.0098	0.1850	-0.0160	9

A: Simple SMC without NCS
B: Simple SMC in NCS
C: SMC based FGA
D: ANSMC based FGA

NCS, SMC based on FGA and ANSMC based on FGA, a comparative analysis can be performed based on the results obtained by these methods by means of some items such as reaching time, average error, maximum value of control input's amplitude and MSE The compared items for these methods have been presented in Table 1. It can be seen that the performance of control by using the proposed ANSMC method has been considerably improved compared with the other methods.

7. CONCLUSIONS

In this paper a neuro-sliding mode controller based on GA has been proposed to control NCS with packet dropouts in controller-actuator path and has compensated adverse influences of packet dropouts in NCS. AFGA has been introduced that improves the population, crossover and mutation probability of simple GA. Sliding function parameters have been tuned using FGA and simple GA and their performances have been compared. Also, the performance of SMC with constant parameters for sliding function has been considered and compared with GA and FGA results. The simulation results show that the FGA has improved the population, convergence speed and the dynamic performance of the control object obviously. Consequently, FGA is better than simple GA and constant parameters, and therefore SMC based on FGA can be used in online control applications in which the quick control action is required. Next an ANSMC based on RBF neural network approximation, has been represented to eliminate chattering phenomenon in SMC. This controller has been applied for a numerical example and the results illustrate that ANSMC has provided high-performance dynamic characteristics and is robust with regard to plant parameter variations and external disturbances in NCS with packet dropouts.

Although the proposed control scheme has been applied to a linear system, due to the applicability of SMC and neural networks, the proposed method can be easily applied to nonlinear systems in presence of uncertainties and disturbances.

8. REFERENCES

1. Ling, Q. and Lemmon, M. D., "Soft real-time scheduling of networked control systems with dropouts governed by a markov chain", in American Control Conference, Proceedings of the 2003, IEEE. Vol. 6, (2003), 4845-4850.
2. Wang, Z., Ho, D. W. and Liu, X., "Variance-constrained control for uncertain stochastic systems with missing measurements", ***Systems, Man and Cybernetics, Part A: Systems and Humans, IEEE Transactions on***, Vol. 35, No. 5, (2005), 746-753.
3. Xiong, J. and Lam, J., "Stabilization of linear systems over networks with bounded packet loss", ***Automatica***, Vol. 43, No. 1, (2007), 80-87.
4. Hu, S. and Yan, W.-Y., "Stability robustness of networked control systems with respect to packet loss", ***Automatica***, Vol. 43, No. 7, (2007), 1243-1248.
5. Li, J., Yanhong, X., Haibin, Y. and Zhang, Q., "Modeling and control of networked control systems with bounded delay and packet disordering", in Control and Decision Conference (CCDC), 2010 Chinese, IEEE, (2010), 3122-3127.
6. Zhao, D., Li, C. and Ren, J., "Fuzzy speed control and stability analysis of a networked induction motor system with time delays and packet dropouts", ***Nonlinear Analysis: Real World Applications***, Vol. 12, No. 1, (2011), 273-287.
7. Yue, D., Han, Q.-L. and Lam, J., "Network-based robust H∞ control of systems with uncertainty", ***Automatica***, Vol. 41, No. 6, (2005), 999-1007.
8. Fleming, P. J. and Purshouse, R. C., "Evolutionary algorithms in control systems engineering: A survey", ***Control Engineering Practice***, Vol. 10, No. 11, (2002), 1223-1241.
9. Yu, K.-W. and Hu, S.-C., "An application of ac servo motor by using particle swarm optimization based sliding mode controller", in Systems, Man and Cybernetics, International Conference on, IEEE, Vol. 5, (2006), 4146-4150.
10. Chen, Z.-m., Meng, W.-j., Zhang, J.-g. and Zeng, J.-c., "Scheme of sliding mode control based on modified particle swarm optimization", ***Systems Engineering-Theory & Practice***, Vol. 29, No. 5, (2009), 137-141.
11. Vaez Zadeh, S. and Zamanian, M., "Permanent magnet DC motor sliding mode control system", ***International Journal of Engineering-Transactions A: Basics***, Vol. 16, No. 4, (2003), 367-376.
12. Soltani, J. and Zarchi, H. A., "Robust optimal speed tracking control of a current sensorless synchronous reluctance motor drive using a new sliding mode controller", in Power Electronics and Drive Systems, The Fifth International Conference on, IEEE, Vol. 1, (2003), 474-479.
13. Shtessel, Y., Edwards, C., Fridman, L. and Levant, A., "Sliding mode control and observation, Springer, (2014).
14. Perruquetti, W. and Barbot, J.-P., "Sliding mode control in engineering, CRC Press, (2002).
15. Bandyopadhyay, B. and Janardhanan, S., "Discrete-time sliding mode control", Berlin, (2006).
16. Gao, W., Wang, Y. and Homaifa, A., "Discrete-time variable structure control systems", ***Industrial Electronics, IEEE Transactions on***, Vol. 42, No. 2, (1995), 117-122.
17. Meng, X. and Song, B., "Fast genetic algorithms used for pid parameter optimization", in Automation and Logistics, IEEE International Conference on, (2007), 2144-2148.
18. Austina, N., Kumarb, P. S. and Kanthavelkumaranc, N., "Artificial neural network involved in the action of optimum mixed refrigerant (domestic refrigerator)", ***International Journal of Engineering***, Vol. 26, No. 10, (2013), 1025-2495.

Designing a Neuro-Sliding Mode Controller for Networked Control Systems with Packet Dropout

M. H. Vali, B. Rezaie, Z. Rahmani

Facultyof Electrical and Computer Engineering, Babol Noshirvani University of Technology, Babol, Iran

PAPER INFO

Paper history:
Received 18 February 2016
Received in revised form 29 March 2016
Accepted 14 April 2016

Keywords:
Networked Control Systems
Packet Dropouts
Sliding Mode Control
Genetic Algorithm
Radial-basis Function Neural Network

چکیده

در این مقاله طراحی کنترل در سیستم‌های کنترل شبکه‌ای با وجود حذف تصادفی بسته‌های داده در مسیر کنترل‌کننده به محرک بررسی می‌گردد. در این راستا حذف تصادفی بسته‌های داده با روش متغیرهای تصادفی مستقل با توزیع باینری برنولی مدل‌سازی شده است. برای جبران اثرات نامطلوب حذف تصادفی بسته‌های داده در سیستم‌های کنترل شبکه‌ای از یک کنترل‌کننده مود لغزشی استفاده شده است. ابتدا برای تعیین پارامترهای تابع سوئیچینگ در کنترل مود لغزشی از یک روش بهبود یافته از الگوریتم ژنتیک استفاده شده است. این الگوریتم بهبود یافته، همگرایی سریع‌تر همراه با عملکرد دینامیکی مطلوب‌تری را نسبت به الگوریتم ژنتیک معمولی به‌ویژه در کاربردهای آنلاین فراهم می‌سازد. سپس، یک روش تطبیقی کنترل مود لغزشی عصبی مبتنی بر تقریب با شبکه‌های عصبی با تابع پایه شعاعی به منظور حذف پدیدهٔ چترینگ پیشنهاد شده است. برای نمایش کارایی کنترل‌کننده پیشنهادی در سیستم های کنترل شبکه‌ای یک مثال عددی آورده شده است. نتایج حاصل شده نشان‌دهنده بهبود عملکرد دینامیکی سیستم و مقاومت در برابر تغییرات پارامترهای سیستم و اغتشاشات خارجی می‌باشد.

***doi**: 10.5829/idosi.ije.2016.29.04a.07*