DOI: 10.7641/CTA.2018.70791

离散非线性零和博弈的事件驱动最优控制方案

张 欣[1†], 薄迎春[1], 崔黎黎[2]

(1. 中国石油大学(华东) 信息与控制工程学院, 山东 青岛 266580;

2. 沈阳师范大学 科信软件学院, 辽宁 沈阳 110034)

摘要: 在求解离散非线性零和博弈问题时, 为了在有效降低网络通讯和控制器执行次数的同时保证良好的控制效果, 本文提出了一种基于事件驱动机制的最优控制方案. 首先, 设计了一个采用新型事件驱动阈值的事件驱动条件, 并根据贝尔曼最优性原理获得了最优控制对的表达式. 为了求解该表达式中的最优值函数, 提出了一种单网络值迭代算法. 利用一个神经网络构建评价网. 设计了新的评价网权值更新规则. 通过在评价网、控制策略及扰动策略之间不断迭代, 最终获得零和博弈问题的最优值函数和最优控制对. 然后, 利用Lyapunov稳定性理论证明了闭环系统的稳定性. 最后, 将该事件驱动最优控制方案应用到了两个仿真例子中, 验证了所提方法的有效性.

关键词: 博弈论; 事件驱动; 自适应动态规划; 最优控制

引用格式: 张欣, 薄迎春, 崔黎黎. 离散非线性零和博弈的事件驱动最优控制方案. 控制理论与应用, 2018, 35(5): 619 – 626

中图分类号: TP273 **文献标识码**: A

Event-triggered optimal control scheme for discrete-time nonlinear zero-sum games

ZHANG Xin[1†], BO Ying-chun[1], CUI Li-li[2]

(1. College of Information and Control Engineering, China University of Petroleum, Qingdao Shandong 266580, China;
2. Sofeware College, Shenyang Normal University, Shenyang Liaoning 110034, China)

Abstract: In order to reduce the network communication and controller execution frequency while guarantee a desired control performance, an event-triggered optimal control scheme is proposed for solving the optimal control pair of discrete-time nonlinear zero-sum games in this paper. Firstly, an event-triggered condition with new event-triggered threshold is designed. The expression of the optimal control pair is obtained based on the Bellman optimality principle. Then, a single network value iteration algorithm is proposed to solve the optimal value function in this expression. A neural network is used to construct the critic network. Novel weight update rule of the critic network is derived. Through the iteration between the critic network, the control policy and the disturbance policy, the optimal value function and the optimal control pair can be solved. Further, the Lyapunov theory is used to prove the stability of the event-triggered closed-loop system. Finally, the event-triggered optimal control mechanism is applied to two examples to verify its effectiveness.

Key words: game theory; event-triggered; adaptive dynamic programming; optimal control

Citation: ZHANG Xin, BO Yingchun, CUI Lili. Event-triggered optimal control scheme for discrete-time nonlinear zero-sum games. *Control Theory & Applications*, 2018, 35(5): 619 – 626

1 引言(Introduction)

近年来, 零和博弈问题在博弈论领域和最优控制领域获得了广泛关注[1–3]. 这是由于零和博弈具有两个决策者, 一方面要求控制输入使性能指标取极小, 而在干扰影响较大时, 又必须考虑干扰信号使性能指标取极大. 这样的对抗性设计既能保证系统在取最优性的同时又具有较好的抗干扰能力. 然而现有的求解零和博弈问题的方法大都采用时间驱动机制, 即控制器是连续更新的, 在每一个采样时刻系统状态与控制器之间都要进行数据通讯, 控制输入都需要计算并执行. 这就大大增加了通讯网络和执行器的负担.

与传统的采样方法不同, 事件驱动机制采用一种

收稿日期: 2017−11−01; 录用日期: 2018−01−23.

†通信作者. E-mail: zhangxin@upc.edu.cn; Tel.: +86 15564879644.

本文责任编委: 魏韡.

山东省自然科学基金项目(BS2015DX009), 国家自然科学基金项目(61703289)资助.

Supported by the National Natural Science Foundation of Shandong Province (BS2015DX009) and the National Natural Science Foundation of China (61703289).

非周期采样模式[4–7]. 文献[4]证明了这种非周期采样比周期采样在计算方面更加有利. 事件驱动机制预先设定了一个事件驱动条件, 只有当该条件不被满足时, 才对系统状态进行采样, 更新系统的控制输入, 在两次更新之间采用零阶保持器保证控制器的输出. 因此, 能够有效地降低网络通讯和控制器执行次数, 同时还能保证系统具有良好的控制性能. 文献[5]研究了线性系统的事件驱动控制. 文献[6]设计了事件驱动光电跟踪系统. Shaoo等人在文献[7]中研究了连续非线性系统的事件驱动状态反馈控制方案. 文献[8]将事件驱动控制带入到了最优控制领域. 事件驱动控制在求解连续系统的零和博弈问题方面也有了相应的成果, 文献[9]将H_∞问题转化为零和博弈问题, 然后基于事件驱动机制进行求解. 据笔者所知, 目前还没有文献利用事件驱动机制求解离散非线性系统的零和博弈问题.

离散非线性系统的零和博弈问题需要求解离散Hamilton-Jacobi-Isaacs (HJI)方程来获得Nash平衡点, 即最优控制对. 但是对于非线性系统来说, HJI方程的解析解很难获得. Werbos在文献[10]中提出了一种有效的求解最优控制问题的方法—自适应动态规划(adaptive dynamic programming, ADP) 算法, 并且得到了广泛应用[11–13]. 文献[11]利用ADP算法处理鲁棒近似最优跟踪问题. 王鼎等人在文献[12]中综述了连续时间非线性系统的自适应评判鲁棒控制设计的最新研究成果. 文献[13]研究了离散非线性系统的事件驱动控制问题. ADP算法自其诞生之日起产生了一系列的同义词, 例如: 自适应评价设计、启发式动态规划、近似动态规划、神经元动态规划和增强学习等等. 2006年在美国科学基金会组织的“2006 NSF Workshop and Outreach Tutorials on Approximate Dynamic Programming”研讨会上, 建议将该方法统称为“adaptive/approximate dynamic programming (自适应/近似动态规划)”. ADP算法已经在一些文献中被用来处理零和博弈问题, 并取得了一定的理论研究成果[14–17]. 然而这些研究都是基于时间驱动机制进行的.

本文将事件驱动机制、ADP算法和神经网络各自优势相结合, 提出了一种求解离散非线性零和博弈问题的事件驱动单网络值迭代控制方案. 首先设计了一个新型的事件驱动阈值. 根据贝尔曼最优性原理获得了最优控制对表达式. 然而, 由于HJI固有的非线性其解析解难以获得, 导致该最优控制对无法直接求解. 因此, 一种单网络值迭代算法被提出. 只利用一个神经网络构建评价网, 从而代替了典型ADP算法中的评价—控制双网结构, 有效减少了神经网络的训练次数. 然后, 根据HJI方程和梯度下降法设计了评价网的权值更新规则. 接着, 利用Lyapunov稳定性理论证明了闭环系统的稳定性. 最后, 将事件驱动最优控制方案应用到了两个仿真例子中, 验证了所提方案既能够有效地降低网络通讯和控制器执行次数, 减少神经网络的训练次数, 又能够保证具有良好的性能.

2 问题描述(Problem descriptions)

考虑如下离散非线性系统的零和博弈问题, 其状态方程描述为

$$\boldsymbol{x}_{k+1} = f(\boldsymbol{x}_k) + g(\boldsymbol{x}_k)\boldsymbol{u}_k + h(\boldsymbol{x}_k)\boldsymbol{w}_k, \tag{1}$$

相应的性能指标函数为普通二次型形式

$$J(\boldsymbol{x}_0) = \sum_{k=0}^{\infty} U(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{w}_k), \tag{2}$$

其中: $\boldsymbol{x}_k \in \Omega \subseteq \mathbb{R}^n$为状态向量; $\boldsymbol{u}_k \in \mathbb{R}^{m_1}$为控制输入, 控制目标是使得性能指标函数最小, 而扰动输入$\boldsymbol{w}_k \in \mathbb{R}^{m_2}$则希望使得性能指标函数最大; $f(\cdot)$, $g(\cdot)$和$h(\cdot)$为光滑可微函数; $\boldsymbol{x}_0$为系统初始状态; $U(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{w}_k) = \boldsymbol{x}_k^{\mathrm{T}} Q \boldsymbol{x}_k + \boldsymbol{u}_k^{\mathrm{T}} R \boldsymbol{u}_k - \boldsymbol{w}_k^{\mathrm{T}} S \boldsymbol{w}_k$是对应的效用函数, 矩阵$Q$, R和S是具有适当维数的对称正定矩阵.

假设 1 系统(1)是可控的, 即存在连续控制策略能够渐近镇定系统(1), $f(0) = 0$, $\boldsymbol{x}_k = 0$是系统(1)唯一的平衡点[17].

假设 2 $f + gu + hw$在紧集$\Omega \subseteq \mathbb{R}^n$上李普希兹连续[17].

定义 1 容许控制是指控制输入$\boldsymbol{u}_k$在紧集$\Omega \subseteq \mathbb{R}^{m_1}$上连续且$\boldsymbol{u}(0) = 0$, 能够控制系统(1)稳定并且保证性能指标函数(2)有界, $\forall \boldsymbol{x}_0 \in \Omega$[17].

由容许控制$\boldsymbol{u}_k$和扰动输入$\boldsymbol{w}_k$定义值函数

$$V(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{w}_k) = \sum_{n=k}^{\infty} \{\boldsymbol{x}_n^{\mathrm{T}} Q \boldsymbol{x}_n + \boldsymbol{u}_n^{\mathrm{T}} R \boldsymbol{u}_n - \boldsymbol{w}_n^{\mathrm{T}} S \boldsymbol{w}_n\}. \tag{3}$$

求解由式(1)–(2)描述的离散非线性系统的零和博弈问题的最优控制对, 要求最优值函数满足

$$V^*(\boldsymbol{x}_k) = \min_{\boldsymbol{u}_k} \max_{\boldsymbol{w}_k} V(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{w}_k) = \max_{\boldsymbol{w}_k} \min_{\boldsymbol{u}_k} V(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{w}_k). \tag{4}$$

根据Bellman最优性原理, 最优值函数$V^*(\boldsymbol{x}_k)$满足离散HJI方程[16]

$$V^*(\boldsymbol{x}_k) = U(\boldsymbol{x}_k, \boldsymbol{u}_k^*, \boldsymbol{w}_k^*) + V^*(\boldsymbol{x}_{k+1}), \tag{5}$$

其中最优控制对$(\boldsymbol{u}_k^*, \boldsymbol{w}_k^*)$应该满足

$$\begin{cases} \dfrac{\partial H(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{w}_k)}{\partial \boldsymbol{u}_k} = 0, \\ \dfrac{\partial H(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{w}_k)}{\partial \boldsymbol{w}_k} = 0, \end{cases} \tag{6}$$

$H(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{w}_k)$为汉密尔顿函数

$$H(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{w}_k) = U(\boldsymbol{x}_k, \boldsymbol{u}_k, \boldsymbol{w}_k) + \boldsymbol{\lambda}_{k+1}(f(\boldsymbol{x}_k) + g(\boldsymbol{x}_k)\boldsymbol{u}_k + h(\boldsymbol{x}_k)\boldsymbol{w}_k), \tag{7}$$

其中协状态$\boldsymbol{\lambda}_{k+1}=\dfrac{\partial V^*(\boldsymbol{x}_{k+1})}{\partial \boldsymbol{x}_{k+1}}$. 因此,

$$\boldsymbol{u}^*(\boldsymbol{x}_k)=-\frac{1}{2}R^{-1}g^{\mathrm{T}}(\boldsymbol{x}_k)\boldsymbol{\lambda}_{k+1}, \tag{8a}$$

$$\boldsymbol{w}^*(\boldsymbol{x}_k)=\frac{1}{2}S^{-1}h^{\mathrm{T}}(\boldsymbol{x}_k)\boldsymbol{\lambda}_{k+1}. \tag{8b}$$

3 事件驱动最优控制方案 (Event-triggered optimal control mechanism)

3.1 事件驱动条件(Event-triggered condition)

在事件驱动机制中, 定义$\{k_i\}_0^\infty$是一个单调递增序列, k_i代表第i个采样时刻, $i=0,1,2,3,\cdots$. 这个采样系统的输出是由系统(1)在k_i时刻的状态$\boldsymbol{x}_{k_i}$组成的序列. 定义事件驱动误差为

$$\boldsymbol{e}_k=\boldsymbol{x}_{k_i}-\boldsymbol{x}_k,\ k\in[k_i,k_{i+1}). \tag{9}$$

事件驱动条件为

$$\|\boldsymbol{e}_k\|\leqslant e_{\mathrm{T}}, \tag{10}$$

其中e_{T}为事件驱动阈值. 只有当$\|\boldsymbol{e}_k\|>e_{\mathrm{T}}$时, 驱动条件不再满足, 系统进行采样. 事件驱动误差重置为零, $\boldsymbol{e}_{k_i}=0$. 反馈控制输入$\boldsymbol{u}(\boldsymbol{x}_{k_i})=\boldsymbol{\mu}(\boldsymbol{x}_{k_i})$更新, 并且通过零阶保持器, 该控制输入在$k\in[k_i,k_{i+1})$时间段内保持不变$\boldsymbol{u}(\boldsymbol{x}_k)=\boldsymbol{\mu}(\boldsymbol{x}_{k_i})$, 直到下一个采样时刻. 需要注意的是, 在本文中假设事件驱动只对控制器$\boldsymbol{u}_k$有影响, 而对扰动输入$\boldsymbol{w}_k$没有影响. 根据式(9), 可得

$$\boldsymbol{u}(\boldsymbol{x}_k)=\boldsymbol{\mu}(\boldsymbol{e}_k+\boldsymbol{x}_k). \tag{11}$$

因此, 系统状态方程(1)重写为

$$\begin{aligned}\boldsymbol{x}_{k+1}=\ & f(\boldsymbol{x}_k)+g(\boldsymbol{x}_k)\boldsymbol{\mu}(\boldsymbol{e}_k+\boldsymbol{x}_k)+\\ & h(\boldsymbol{x}_k)\boldsymbol{w}_k.\end{aligned} \tag{12}$$

在事件驱动机制中, 控制输入只在采样时刻更新, 即只在k_i时刻生成. 因此, 状态反馈控制策略(8a)应该表示为

$$\boldsymbol{\mu}^*(\boldsymbol{x}_k)=-\frac{1}{2}R^{-1}g^{\mathrm{T}}(\boldsymbol{x}_{k_i+1})\boldsymbol{\lambda}_{k_i+1},\ k\in[k_i,k_{i+1}). \tag{13}$$

假设 3　存在正数L, 满足[13]

$$\|\boldsymbol{x}_{k+1}\|\leqslant L\|\boldsymbol{e}_k\|+L\|\boldsymbol{x}_k\|. \tag{14}$$

当最后一次采样时刻为k_i, $k\in[k_i,k_{i+1})$, 根据式(9), 可得$\boldsymbol{e}_{k+1}=\boldsymbol{x}_{k_i}-\boldsymbol{x}_{k+1}$. 显然

$$\begin{aligned}\|\boldsymbol{e}_{k+1}\|\leqslant & \|\boldsymbol{x}_{k_i}\|+\|\boldsymbol{x}_{k+1}\|\leqslant\\ & \|\boldsymbol{x}_{k_i}\|+L\|\boldsymbol{e}_k\|+L\|\boldsymbol{x}_k\|\leqslant\\ & \|\boldsymbol{x}_{k_i}\|+L\|\boldsymbol{e}_k\|+L(\|\boldsymbol{x}_{k_i}\|+\|\boldsymbol{e}_k\|)=\\ & 2L\|\boldsymbol{e}_k\|+(1+L)\|\boldsymbol{x}_{k_i}\|.\end{aligned} \tag{15}$$

利用其递归性可得

$$\begin{aligned}\|\boldsymbol{e}_k\|\leqslant & 2L\|\boldsymbol{e}_{k-1}\|+(1+L)\|\boldsymbol{x}_{k_i}\|\leqslant\\ & (2L)^2\|\boldsymbol{e}_{k-2}\|+[1+2L](1+L)\|\boldsymbol{x}_{k_i}\|\leqslant\\ & \vdots\\ & (2L)^{k-k_i}\|\boldsymbol{e}_{k_i}\|+[1+2L+(2L)^2+\\ & (2L)^3+\cdots+(2L)^{k-k_i-1}]\times\\ & (1+L)\|\boldsymbol{x}_{k_i}\|.\end{aligned} \tag{16}$$

为了确保等比数列收敛, 要求$2L<1$, 即$L<0.5$. 由于在每一个采样时刻$\boldsymbol{e}_{k_i}=0$, 则式(16)变为

$$\|\boldsymbol{e}_k\|\leqslant\frac{(1-(2L)^{k-k_i})(1+L)}{1-2L}\|\boldsymbol{x}_{k_i}\|. \tag{17}$$

定义事件驱动阈值为

$$e_{\mathrm{T}}=\frac{\alpha(1-(2L)^{k-k_i})(1+L)}{1-2L}\|\boldsymbol{x}_{k_i}\|, \tag{18}$$

其中$\alpha\in(0,1]$为常数.

3.2 单网络ADP值迭代算法及神经网络实现(Single network ADP value iteration algorithm and neural network implementation)

对于非线性系统来说, HJI方程(5)的解很难直接求解. 为了获得式(8b)和式(13)中最优值函数的值, 根据贝尔曼最优性原理, 利用ADP值迭代算法来近似求解.

首先, 给定一个初始值函数$V_0(\boldsymbol{x}_k)$, 一般情况选择$V_0(\boldsymbol{x}_k)=0$. $\boldsymbol{u}_0$和$\boldsymbol{w}_0$可以通过下式计算获得:

$$\begin{cases}\boldsymbol{u}_0=\arg\min\limits_{\boldsymbol{u}_k}\{U(\boldsymbol{x}_k,\boldsymbol{u}_0,\boldsymbol{w}_0)+V_0(\boldsymbol{x}_{k+1})\},\\ \boldsymbol{w}_0=\arg\max\limits_{\boldsymbol{w}_k}\{U(\boldsymbol{x}_k,\boldsymbol{u}_0,\boldsymbol{w}_0)+V_0(\boldsymbol{x}_{k+1})\}.\end{cases} \tag{19}$$

那么迭代的值函数$V_1(\boldsymbol{x}_k)$为

$$V_1(\boldsymbol{x}_k)=U(\boldsymbol{x}_k,\boldsymbol{u}_0,\boldsymbol{w}_0)+V_0(\boldsymbol{x}_{k+1}). \tag{20}$$

以此类推, 相应的迭代策略$\boldsymbol{u}_j$和$\boldsymbol{w}_j$迭代规则为

$$\boldsymbol{u}_j=-\frac{1}{2}R^{-1}g^{\mathrm{T}}(\boldsymbol{x}_k)\boldsymbol{\lambda}_{j(k+1)}, \tag{21a}$$

$$\boldsymbol{w}_j=\frac{1}{2}S^{-1}h^{\mathrm{T}}(\boldsymbol{x}_k)\boldsymbol{\lambda}_{j(k+1)}. \tag{21b}$$

值函数$V_{j+1}(x_k)$的迭代规则为

$$\begin{aligned}V_{j+1}(\boldsymbol{x}_k)=\ & (\boldsymbol{x}_k^{\mathrm{T}}Q\boldsymbol{x}_k+\boldsymbol{u}_j^{\mathrm{T}}R\boldsymbol{u}_j-\boldsymbol{w}_j^{\mathrm{T}}S\boldsymbol{w}_j)+\\ & V_j(\boldsymbol{x}_{k+1}),\end{aligned} \tag{22}$$

其中j表示迭代次数.

本文采用的是单神经网络结构, 只利用一个评价网来近似值函数. 该评价网由以下3层神经网络构成:

$$V^*(\boldsymbol{x}_k)=W_{\mathrm{c}}^{*\mathrm{T}}\phi_{\mathrm{c}}(V_{\mathrm{c}}^*\boldsymbol{x}_k)+\varepsilon_{\mathrm{c}k}, \tag{23}$$

其中$W_{\mathrm{c}}^*\in\mathbb{R}^{N_{\mathrm{c}}\times 1}$为未知的隐含层到输出层的理想神经网络权值, $V_{\mathrm{c}}^*\in\mathbb{R}^{N_{\mathrm{c}}\times n}$为输入层到隐含层的理想神经网络权值, N_{c}是隐含层节点数, $\phi_{\mathrm{c}}(\cdot)$为评价网激活函数, $\varepsilon_{\mathrm{c}k}\in\mathbb{R}$为评价网近似误差.

在评价网训练过程中, 输入层到隐含层的权值$V_{\rm c}^*$保持不变. 仅训练隐含层到输出层的权值, 定义$\hat{W}_{\rm c}(j)$为其估计值, 则实际的评价网输出为

$$V_j(\boldsymbol{x}_k)=\hat{W}_{\rm c}^{\rm T}(j)\phi_{\rm c}(\bar{x}_k),\tag{24}$$

其中$\bar{x}_k=V_{\rm c}^*\boldsymbol{x}_k$.

根据值函数的迭代规则(22)和评价网输出(24)以及HJI方程(5), 设计评价网的训练误差为

$$\begin{aligned}e_{\rm c}=&U(\boldsymbol{x}_k,\boldsymbol{u}_j,\boldsymbol{w}_j)+V_j(\boldsymbol{x}_{k+1})-V_j(\boldsymbol{x}_k)=\\&U(\boldsymbol{x}_k,\boldsymbol{u}_j,\boldsymbol{w}_j)+\\&\hat{W}_{\rm c}^{\rm T}(j)\phi_{\rm c}(\bar{x}_{k+1})-\hat{W}_{\rm c}^{\rm T}(j)\phi_{\rm c}(\bar{x}_k)=\\&\boldsymbol{x}_k^{\rm T}Q\boldsymbol{x}_k+\boldsymbol{u}_j^{\rm T}R\boldsymbol{u}_j-\boldsymbol{w}_j^{\rm T}S\boldsymbol{w}_j+\\&\hat{W}_{\rm c}^{\rm T}(j)\Delta\phi_{\rm c}(k),\end{aligned}\tag{25}$$

其中: $\Delta\phi_{\rm c}(k)=\phi_{\rm c}(\bar{x}_{k+1})-\phi_{\rm c}(\bar{x}_k)$, $\boldsymbol{u}_j$和$\boldsymbol{w}_j$的值由式(21)计算获得.

定义最小化目标函数为

$$E_{\rm c}=\frac{1}{2}e_{\rm c}^2.\tag{26}$$

利用梯度下降法, 可得评价网的权值更新规则为

$$\begin{aligned}&\hat{W}_{\rm c}(j+1)=\hat{W}_{\rm c}(j)+\Delta\hat{W}_{\rm c}(j),\\&\Delta\hat{W}_{\rm c}(j)=-\alpha_{\rm c}(\frac{\partial E_{\rm c}}{\partial\hat{W}_{\rm c}(j)})=\\&\qquad-\alpha_{\rm c}\Delta\phi_{\rm c}(k)(\boldsymbol{x}_k^{\rm T}Q\boldsymbol{x}_k+\boldsymbol{u}_j^{\rm T}R\boldsymbol{u}_j-\\&\qquad\boldsymbol{w}_j^{\rm T}S\boldsymbol{w}_j+\hat{W}_{\rm c}^{\rm T}(j)\Delta\phi_{\rm c}(k))^{\rm T},\end{aligned}\tag{27}$$

其中$\alpha_{\rm c}$为评价网学习率.

假设 4 存在常数θ,α,β满足

$$0\leqslant V^*(\boldsymbol{x}_{k+1})\leqslant\theta U(\boldsymbol{x}_k,\boldsymbol{u}_k,\boldsymbol{w}_k),\tag{28}$$

$$0\leqslant\eta_1V^*(\boldsymbol{x}_k)\leqslant V_0(\boldsymbol{x}_k)\leqslant\eta_2V^*(\boldsymbol{x}_k),\tag{29}$$

其中: $0<\theta<\infty$, $0<\eta_1<1$, $1\leqslant\eta_2<\infty$, V_0为任意初始值函数[17].

若假设4成立, 当迭代次数j趋于无穷大时, $V_j(\boldsymbol{x}_k)$将收敛到最优值函数$V^*(\boldsymbol{x}_k)$, 控制对$(\boldsymbol{u}_j,\boldsymbol{w}_j)$收敛到最优控制对$(\boldsymbol{u}^*,\boldsymbol{w}^*)$. 评价网权值$\hat{W}_{\rm c}(j)$收敛到$W_{\rm c}$, $W_{\rm c}=\lim\limits_{j\to\infty}\hat{W}_{\rm c}(j)$[17]. 为了避免神经网络权值在训练过程中陷入到局部极小值, 在训练中需要加入持续激励信号.

注 1 根据假设2, $f+gu+hw$是李普希兹连续的. 并且有限的控制输入不可能使得系统状态在一步之内跳变到无穷大, 因此$f(\boldsymbol{x}_k)+g(\boldsymbol{x}_k)\boldsymbol{u}_k+h(\boldsymbol{x}_k)\boldsymbol{w}_k$是有限的. 考虑到$V^*(\boldsymbol{x}_k)$对于任意有限的系统状态和控制输入都是有限的, 因此一定存在$0<\theta<\infty$能够保证不等式(28)成立. 此外, 由于任意的初始值函数$V_0(\boldsymbol{x}_k)$是有界的, 那么不等式(29)也很容易得到满足.

注 2 与典型的ADP算法不同, 本文采用的是单网络结构, 只利用一个评价网来近似值函数, 省略掉了用来近似控制策略和扰动策略的两个控制网. 由于本文研究的是模型完全已知仿射非线性系统, 因而模型网也被省略. 系统状态方程具有的仿射结构保证了控制策略和扰动策略可以根据最优性原理直接通过计算获得. 如果系统模型未知或者是非仿射结构, 可以通过增加模型网来构建仿射结构的系统状态方程.

单网络ADP值迭代算法具体执行步骤如下:

步骤 1 初始化参数Q, R, S, ξ, $\alpha_{\rm c}$, $j_{\max}$, 神经网络权值$V_{\rm c}^*$和$\boldsymbol{x}_k$;

步骤 2 令$j=0$, $\hat{W}_{\rm c}(0)=0$, 使得$V_0(\boldsymbol{x}_k)=0$;

步骤 3 根据式(19)计算$\boldsymbol{u}_0$和$\boldsymbol{w}_0$;

步骤 4 令$j=j+1$;

步骤 5 根据式(12)计算$\boldsymbol{x}_{k+1}$;

步骤 6 根据式(27)更新权值$\hat{W}_{\rm c}(j+1)$;

步骤 7 根据式(24)计算$V_{j+1}(\boldsymbol{x}_k)$;

步骤 8 根据式(21)计算$\boldsymbol{u}_j$和$\boldsymbol{w}_j$;

步骤 9 如果$\|V_{j+1}(\boldsymbol{x}_k)-V_j(\boldsymbol{x}_k)\|<\xi$或者迭代次数$j>j_{\max}$, 跳转步骤10, 否则跳转步骤4;

步骤 10 近似最优的控制对已获得, 算法结束.

3.3 事件驱动单网络值迭代算法(Event-triggered single network value iteration algorithm, ET-SNVI)

根据第3.1节可知, 事件驱动阈值为$e_{\rm T}$, 事件驱动条件为$\|\boldsymbol{e}_k\|\leqslant e_{\rm T}$. 当驱动条件不再满足时, 事件驱动误差被重置为零, 控制输入$\boldsymbol{\mu}^*(\boldsymbol{x}_{k_i})$更新. 控制输入和扰动输入的计算公式如式(13)和式(8b)所示, 其中的最优值函数$V^*(\boldsymbol{x}_k)$可通过第3.2节中的单网络值迭代算法逼近. 因此, 最终获得了基于事件驱动的零和博弈问题的近似最优解为

$$\boldsymbol{\mu}(\boldsymbol{x}_k)=-\frac{1}{2}R^{-1}g^{\rm T}(\boldsymbol{x}_{k_i})\boldsymbol{\lambda}_{k_i+1},\ k\in[k_i,k_{i+1}),\tag{30a}$$

$$\boldsymbol{w}(\boldsymbol{x}_k)=\frac{1}{2}S^{-1}h^{\rm T}(\boldsymbol{x}_k)\boldsymbol{\lambda}_{k+1},\tag{30b}$$

其中协状态$\boldsymbol{\lambda}_{k_i+1}x$和$\boldsymbol{\lambda}_{k+1}$中的最优值函数由评价网的输出近似$V^*(\boldsymbol{x}_k)=W_{\rm c}^{\rm T}\phi_{\rm c}(V_{\rm c}^*\boldsymbol{x}_k)$.

假设 5 存在正常数α, β和L_1, K_∞类函数α_1和α_2能够使得下列不等式满足[13]:

$$\alpha_1(\|\boldsymbol{x}_k\|)\leqslant V(\boldsymbol{x}_k)\leqslant\alpha_2(\|\boldsymbol{x}_k\|),\tag{31}$$

$$V(\boldsymbol{x}_{k+1})-V(\boldsymbol{x}_k)\leqslant-\alpha V(\boldsymbol{x}_k)+\beta\|\boldsymbol{e}_k\|,\tag{32}$$

$$\alpha_1{}^{-1}(\|\boldsymbol{x}_k\|)\leqslant L_1\|\boldsymbol{x}_k\|.\tag{33}$$

定理 1 对于离散系统(12), 如果假设5成立, 对于$k\in[k_i,k_{i+1})$, $i=0,1,\cdots$, 满足下列不等式:

$$\varphi(k)-\alpha\sigma(k)<0,\tag{34}$$

其中:

$$\sigma(k)=(1-\alpha)^{k-k_i}+(1-(1-\alpha)^{k-k_i})\varphi(k)/\alpha,$$
$$\varphi(k)=(\beta(1-(2L)^{k-k_i})(1+L)L_1)/(1-2L).$$

则系统(12)是渐近稳定的.

证　由式(33)可知

$$\|\boldsymbol{x}_{k_i}\|\leqslant\alpha_1(V(\boldsymbol{x}_{k_i}))\leqslant L_1V(\boldsymbol{x}_{k_i}).\tag{35}$$

将式(18)和式(35)代入到式(32)中, 可得

$$V(\boldsymbol{x}_{k+1})-V(\boldsymbol{x}_k)\leqslant-\alpha V(\boldsymbol{x}_k)+\varphi(k)V(\boldsymbol{x}_{k_i}).\tag{36}$$

求解式(36), 可得

$$V(\boldsymbol{x}_k)\leqslant\sigma(k)V(\boldsymbol{x}_{k_i}).\tag{37}$$

将式(37)代入式(36), 可得

$$V(\boldsymbol{x}_{k+1})-V(\boldsymbol{x}_k)\leqslant(\varphi(k)-\alpha\sigma(k))V(\boldsymbol{x}_{k_i}).\tag{38}$$

应用式(31), 可得

$$\Delta V=V(\boldsymbol{x}_{k+1})-V(\boldsymbol{x}_k)\leqslant(\varphi(k)-\alpha\sigma(k))\alpha_2(\|\boldsymbol{x}_{k_i}\|).\tag{39}$$

因此, 当不等式(34)成立时, $\Delta V<0$. 根据Lyapunov稳定性理论系统(12)渐近稳定.　　证毕.

本文提出的事件驱动最优控制方案结构图如图1所示, 其具体步骤如下:

步骤 1　初始化参数α, L, ϵ和$i_{\max}$. 令$i=0$, $k=0$;

步骤 2　根据式(9)和式(18)计算事件驱动误差$\boldsymbol{e}_k$和阈值e_{T};

步骤 3　判断$\|\boldsymbol{e}_k\|$是否大于e_{T}, 如果大于执行步骤4, 如果小于等于跳转步骤6;

步骤 4　$i=i+1$, $\boldsymbol{x}_{k_i}=\boldsymbol{x}_k$, $\boldsymbol{e}_k=0$;

步骤 5　根据式(30a)计算$\boldsymbol{\mu}(\boldsymbol{x}_k)$;

步骤 6　根据式(30b)计算$\boldsymbol{w}(\boldsymbol{x}_k)$;

步骤 7　根据式(12)计算$\boldsymbol{x}_{k+1}$;

步骤 8　如果$\|\boldsymbol{x}_{k+1}-\boldsymbol{x}_k\|\leqslant\epsilon$, 或者$i>i_{\max}$, 跳转步骤9, 否则跳转步骤2;

步骤 9　算法结束.

注 3　将值函数$V(\boldsymbol{x}_k)$定义为系统的李雅普诺夫函数. 根据HJI方程(5)和公式(22), 值函数$V(\boldsymbol{x}_k)$可以表述为系统状态$\boldsymbol{x}_k$的相关函数. 如果系统是一个线性系统, 值函数$V(\boldsymbol{x}_k)=\boldsymbol{x}_k^{\mathrm{T}}P\boldsymbol{x}_k$, 其中$P$为黎卡提方程的解. 显然, 其满足假设5中的不等式(31). 当系统为一个非线性系统的时候, 用评价网$W_{\mathrm{c}}^{\mathrm{T}}\phi_{\mathrm{c}}(\bar{x}_{k+1})$来逼近$V(\boldsymbol{x}_k)$. 适当的选择激活函数$\phi_{\mathrm{c}}(\cdot)$也能够保证不等式(31)成立.

注 4　本文提出的事件驱动单网络值迭代算法是一种离线的算法, 通过在评价网、控制策略和扰动策略之间的不断迭代, 最终获得全局最优控制对, 该最优控制对可以在线直接应用在每一个事件驱动时刻. 而且该算法一般取初始迭代值函数$V_0(\boldsymbol{x}_k)=0$, 不要求提供一个初始稳定增益. 这对非线性系统来说是非常重要的, 因为非线性系统的初始稳定增益并不容易获得.

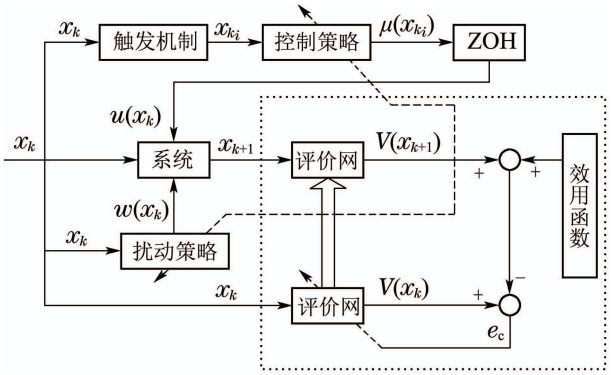


图 1　事件驱动最优控制方案结构图

Fig. 1　The structure of the event-triggered optimal control scheme

4　仿真验证(Simulation)

为验证本文所提的事件驱动最优控制方案的有效性, 本小节将该方案应用到了F–16战斗机和一个非线性系统的仿真例子中.

例 1　F–16战斗机.

考虑如下的F–16战斗机的离散数学模型[16]:

$$\boldsymbol{x}_{k+1}=F\boldsymbol{x}_k+G\boldsymbol{u}_k+H\boldsymbol{w}_k,\tag{40}$$

其中: $\boldsymbol{x}_k=[\alpha_k\ \ q_k\ \ \delta_{ek}]^{\mathrm{T}}$, α_k为攻击角度, q_k为俯仰角速度, δ_{ek}为升降舵偏转角, u为制动器电压, w为作用到攻击角度上的阵风.

$$F=\begin{bmatrix}0.906488 & 0.0816012 & -0.0005\\ 0.0741349 & 0.90121 & -0.000708383\\ 0 & 0 & 0.132655\end{bmatrix},$$

$$G=\begin{bmatrix}-0.00150808\\ -0.0096\\ 0.867345\end{bmatrix},\ H=\begin{bmatrix}0.00951892\\ 0.00038373\\ 0\end{bmatrix}.$$

性能指标函数如式(2)所示, 其中: $Q\in\mathbb{R}^{3\times3}$, $R\in\mathbb{R}^{1\times1}$和$S\in\mathbb{R}^{1\times1}$为单位阵. 飞行器的初始状态设定为$\boldsymbol{x_0}=[4\ \ 2\ \ 5]^{\mathrm{T}}$. 采用一个3–8–1的3层神经网络来构成评价网, 评价网的初始权值V_{c}在$[-1,1]$之间随机生成. $\hat{W}_{\mathrm{c}}(0)$设定为零, 从而保证初始迭代值函数$V_0(\boldsymbol{x}_k)=0$. 激活函数$\phi_{\mathrm{c}}(\cdot)$选为tansig函数. 评价网学习率$\alpha_{\mathrm{c}}=0.2$. 计算精度为$\xi=10^{-5}$. 评价网训练了2000次, 为了避免神经网络权值陷入局部极小值, 在前800迭代步中加入了持续激励. 评价网权值的收敛轨迹如图2所示.

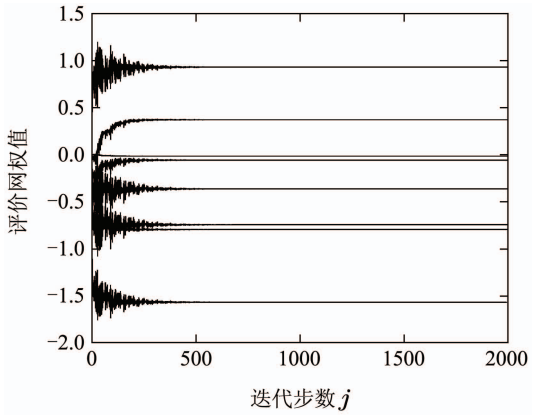


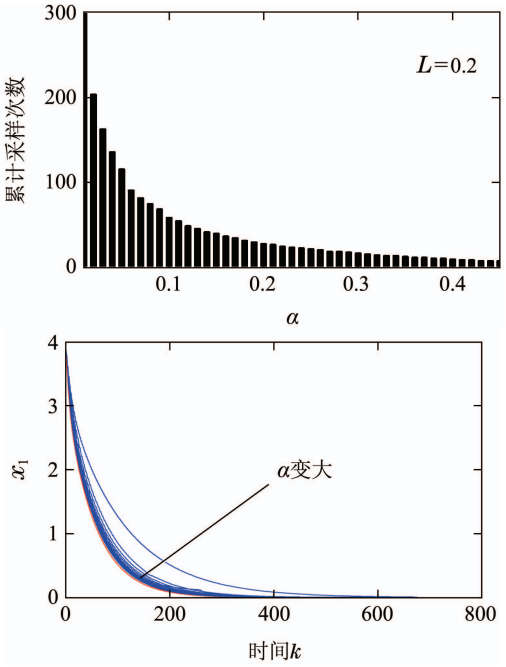
图 2 评价网权值收敛轨迹

Fig. 2 The convergent trajectories of critic network weights

由式(18)可知, 事件驱动阈值e_{T}与α和L的值有关. 为了选择适当的α和L, 作者进行了一系列的试验. 当$L=0.2$时, α取不同的值时, 累计采样次数和系统状态曲线如图3所示. 图中箭头指向的方向为α增大的方向. 从图3中可以看出, 随着α的增大, 累计采样次数逐渐减少, 系统状态x_1和x_2逐渐接近最优状态轨迹. 但是系统状态x_3随着α的增大, 距离最优状态轨迹越来越远. 在综合考虑了累计采样次数和系统性能之后, 最终选择$\alpha=0.1$. 同理, 当$\alpha=0.1$时, 选取不同的L进行了一系列的仿真, 发现随着L的增大, 累计采样次数逐渐减少, 但是对系统状态的影响不大. 最终, 本文选取了$\alpha=0.1$, $L=0.1$来确定事件驱动阈值.

$$e_{\mathrm{T}}=0.1375(1-(0.2)^{k-k_i})\|\boldsymbol{x}_{k_i}\|. \tag{41}$$

当$\alpha=0.1$, $L=0.1$时, 系统的状态轨迹如图4所示. 从图4可以看出, 系统在796步之后能够达到精度$\epsilon=10^{-5}$. 事件驱动误差的范数$\|\boldsymbol{e}_k\|$和阈值e_{T}的变化情况如图5所示.



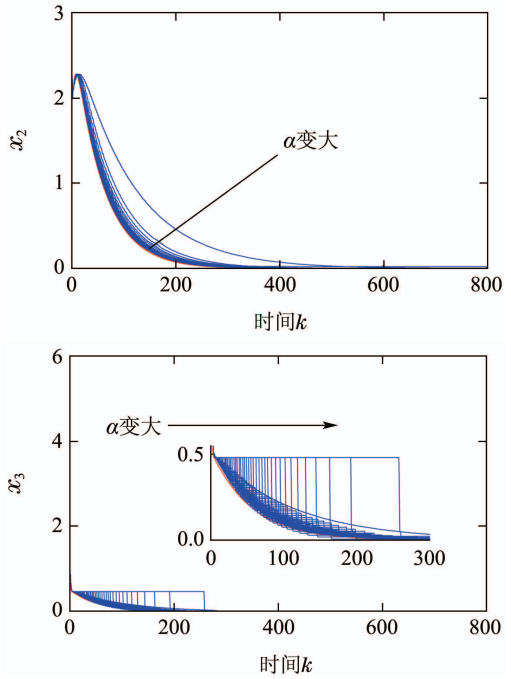


图 3 α取不同值时累计采样次数和系统状态轨迹

Fig. 3 The number of cumulative samples and the trajectories of system states with different α

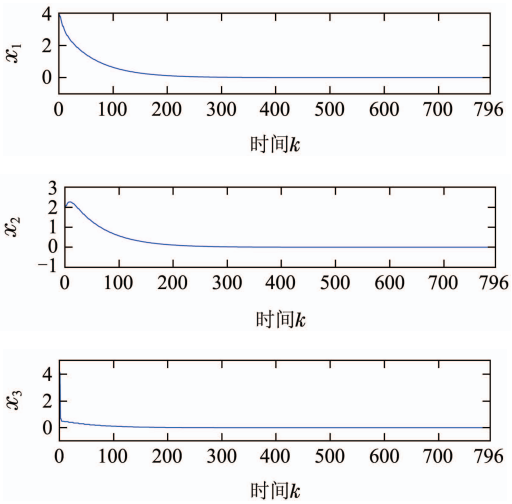


图 4 系统状态轨迹

Fig. 4 The trajectories of system states

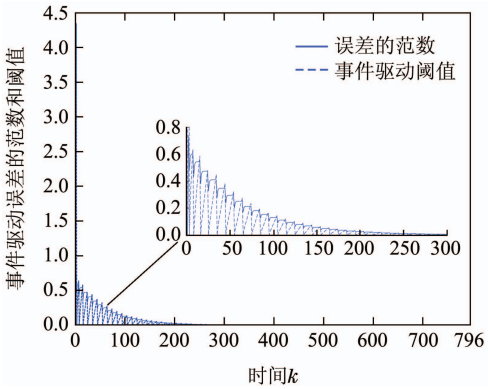


图 5 事件驱动误差的范数和事件驱动阈值轨迹

Fig. 5 The trajectories of the norm of event-triggered error and event-triggered threshold

由于事件驱动条件在前300步变化明显, 所以在图5中给出了前300步的局部放大图. 控制输入和扰动输入的变化轨迹如图6所示. 图7给出了典型ADP算法和事件驱动单网络值迭代算法的累计采样次数对比图.

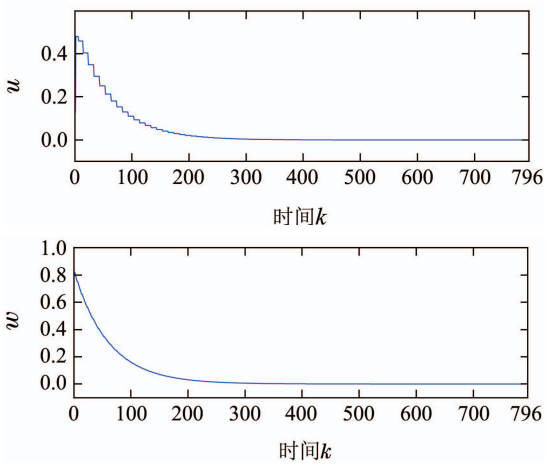


图 6　控制输入和扰动输入轨迹

Fig. 6　The trajectories of control input and distribute input

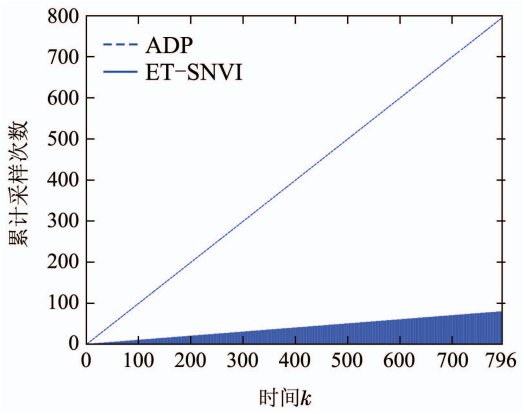


图 7　累计采样次数

Fig. 7　The cumulative samples

如图7所示, 本文所提出的事件驱动单网络值迭代算法只需要进行80次采样, 而典型的时间驱动的ADP算法则需要进行796次采样. 本文所提算法能够减少近90%的通讯次数和计算量. 同时, 由于只采用了一个神经网络, 省略了用来近似控制策略和扰动策略的两个控制网, 所以减少了近67%的神经网络权值训练量.

例 2　离散非线性系统.

考虑如下的离散非线性零和博弈问题, 其状态方程为

$$\boldsymbol{x}_{k+1} = f(\boldsymbol{x}_k) + g(\boldsymbol{x}_k)\boldsymbol{u}_k + h(\boldsymbol{x}_k)\boldsymbol{w}_k, \quad (42)$$

其中:

$$f(\boldsymbol{x}_k) = [f_1(\boldsymbol{x}_k)\ \ f_2(\boldsymbol{x}_k)]^{\mathrm{T}},$$
$$f_1(\boldsymbol{x}_k) = 0.9x_{1k} - 0.1x_{2k},$$
$$f_2(\boldsymbol{x}_k) = 0.9x_{1k} - 0.5x_{2k} + 0.025x_{2k}(\cos x_{1k})^2,$$
$$g(\boldsymbol{x}_k) = [-0.01x_{1k}\ \ 0.5\cos(2x_{1k})]^{\mathrm{T}},$$
$$h(\boldsymbol{x}_k) = [0.1\ \ 0.1\sin(4x_{1k}) + 2]^{\mathrm{T}}.$$

性能指标函数如式(2)所示, 其中Q, R和S为具有适当维数的单位阵. 初始状态设定为$x_0 = [4\ \ 2]^{\mathrm{T}}$. 采用一个2–8–1的3层神经网络来构成评价网, 评价网的初始权值V_{c}在$[-1, 1]$之间随机生成. $\hat{W}_{\mathrm{c}}$设定为零. 激活函数$\phi_{\mathrm{c}}(\cdot)$选为tansig函数. 评价网学习率$\alpha_{\mathrm{c}} = 0.1$. 选取$\alpha = 0.1$, $L = 0.2$来确定事件驱动阈值.

$$e_{\mathrm{T}} = 0.2(1 - (0.4)^{k-k_i})\|\boldsymbol{x}_{k_i}\|. \quad (43)$$

系统的状态轨迹如图8所示. 从图8可以看出, 系统在125步之后能够达到精度$\epsilon = 10^{-5}$. 图9给出了控制输入和扰动输入的变化轨迹. 事件驱动误差的范数$\|\boldsymbol{e}_k\|$和事件驱动阈值e_{T}的变化情况如图10所示. 与典型的时间驱动的ADP算法需要进行125次采样相比, 本文所提的事件驱动最优控制方法只进行了63次采样, 减少了近50%的网络通讯量和控制器计算以及执行次数.

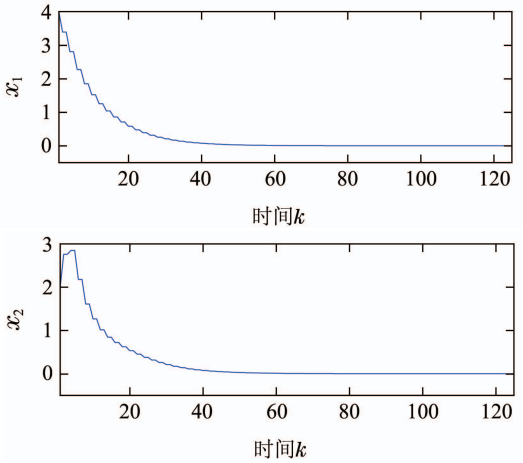


图 8　系统状态轨迹

Fig. 8　The trajectories of system states

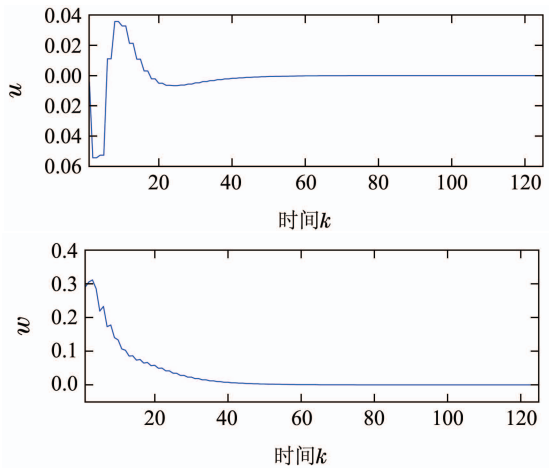


图 9　控制输入和扰动输入轨迹

Fig. 9　The trajectories of control input and distribute input

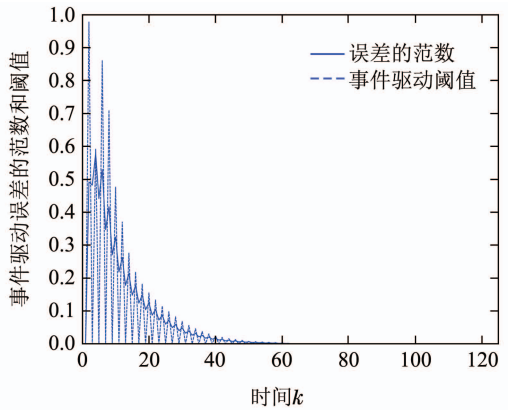


图 10 事件驱动误差的范数和事件驱动阈值的轨迹

Fig. 10 The trajectories of the norm of event-triggered error and event-triggered threshold

从上述仿真结果中可以看出, 本文提出的零和博弈问题的事件驱动最优控制方案, 能够很好的镇定系统, 并且获得零和博弈问题的近似最优控制对. 通过事件驱动机制, 能够有效的减少控制输入与系统之间的数据传输次数、控制器计算次数以及执行器变动次数. 并且单网络值迭代算法能够有效降低神经网络权值的训练量.

5 结论(Conclusions)

本文研究了博弈论中常见的零和博弈问题. 为了降低数据传输和计算次数, 获得最优控制对, 提出了一种基于事件驱动的单网络值迭代算法. 将事件驱动控制应用到零和博弈问题求解中, 设计新型事件驱动阈值. 采用单网络值迭代算法, 利用一个神经网络构建评价网, 根据Bellman最优性原理直接计算控制对, 通过在评价网、控制策略和扰动策略之间进行迭代, 获得最优值函数. 给出了神经网络权训练步骤. 接着, 利用Lyapunov理论证明了闭环系统的稳定性, 并给出了事件驱动最优控制方案的执行步骤. 最后, 将该方案应用于F–16战斗机和一个非线性系统的零和博弈问题仿真实验中, 仿真结果表明所提方法能够获得近似最优控制对, 并且成功地降低了网络通信频率, 控制输入的执行次数以及神经网络权值的训练次数.

参考文献(References):

[1] FU Yue, CHAI Tianyou. Online solution of two-player zero-sum games for linear systems with unknown dynamics [J]. *Control Theory & Applications*, 2015, 32(2): 196 – 201.
(富月, 柴天佑. 具有未知动态的线性系统二人零和博弈问题在线学习方案 [J]. 控制理论与应用, 2015, 32(2): 196 – 201.)

[2] YVES A, PEREZ V. Iterative strategies for solving linearized discrete mean field games systems [J]. *Netw Heterog Media*, 2012, 7(2): 197 – 217.

[3] FU Y, FU J, CHAI T. Robust adaptive dynamic programming of two-player zero-sum games for continuous-time linear systems [J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2015, 26(12): 3314 – 3319.

[4] ASTROM K J, BERNHARDSSON B M. Comparison of Riemann and Lebesgue sampling for first order stochastic systems [C] //*Proceedings of the 41st IEEE Conference on Decision Control*. Las Vegas: IEEE, 2002, 2: 2011 – 2016.

[5] HEEMELES W, DONKERS M, TEEL A. Periodic event-triggered control for linear systems [J]. *IEEE Transactions on Automatic Control*, 2013, 58(4): 847 – 861.

[6] LIANG Yuan, QI Guoqing, LI Yinya, et al. Design and application of event-triggered mechanism for a kind of optical-electronic tracking system [J]. *Control Theory & Applications*, 2017, 34(10): 1328 – 1338.
(梁苑, 戚国庆, 李银伢, 等. 一类光电跟踪系统中事件触发机制的设计及应用 [J]. 控制理论与应用, 2017, 34(10): 1328 – 1338.)

[7] SAHOO A, XU H, JAGANNATHAN S. Neural network-based event-triggered state feedback control of nonlinear continuous-time systems [J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2016, 27(3): 497 – 509.

[8] VAMVOUDAKIS K G. Event-triggered optimal adaptive control algorithm for continuous-time nonlinear systems [J]. *IEEE/CAA Journal of Automatica Sinica*, 2014, 1(3): 282 – 293.

[9] ZHANG Q, ZHAO D, ZHU Y. Event-triggered H_∞ control for continuous-time nonlinear system via concurrent learning [J]. *IEEE Transactions on Systems, Man, and Cybernetics*, 2017, 47(7): 1071 – 1081.

[10] WERBOS P J. Approximate dynamic programming for real-time control and neural modeling [M] //*Handbook of Intelligent Control: Neural, Fuzzy and Adaptive Approaches*. New York: Van Nostrand Reinhold, 1992.

[11] QU Qiuxia, LUO Yanhong, ZHANG Huaguang. Robust approximate optimal tracking control of time-varying trajectory for nonlinear affine systems [J]. *Control Theory & Applications*, 2016, 33(1): 77 – 84.
(屈秋霞, 罗艳红, 张化光. 针对时变轨迹的非线性仿射系统的鲁棒近似最优跟踪控制 [J]. 控制理论与应用, 2016, 33(1): 77 – 84.)

[12] WANG D, HE H, LIU D. Adaptive critic nonlinear robust control: a survey [J]. *IEEE Transactions on Cybernetics*, 2017, 47(10): 3429 – 3451.

[13] DONG L, ZHONG X N, SUN C Y, et al. Adaptive event-triggered control based on heuristic dynamic programming for nonlinear discrete-time systems [J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2017, 28(7): 1594 – 1605.

[14] LUO B, WU H N, HUANG T. Off-policy reinforcement learning for H_∞ control design [J]. *IEEE Transactions on Cybernectics*, 2015, 45(1): 65 – 76.

[15] ZHANG X, ZHANG H G, WANG F Y. A new iteration approach to solve a class of Finite-horizon continuous-time nonaffine nonlinear zero-sum game [J]. *International Journal of Innovative, Computing, Information and Control*, 2011, 7(2): 597 – 608.

[16] AL-TAMIMI A, KHALAF M, LEWIS F L. Adaptive critic designs for discrete-time zero-sum games with application to H_∞ control [J]. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 2007, 37(1): 240 – 247.

[17] LIU D, LI H, WANG D. Neural-network-based zero-sum game for discrete-time nonlinear systems via iterative adaptive dynamic programming algorithm [J]. *Neurocomputing*, 2013, 110(8): 92 – 100.

[18] JIANG Z P, WANG Y. Input-to-state stability for discretetime nonlinear systems [J]. *Automatica*, 2001, 37(6): 857 – 869.

作者简介:

张 欣 (1982–), 女, 讲师, 博士, 目前研究方向为自适应动态规划、最优控制、神经网络等, E-mail: zhangxin@upc.edu.cn;

薄迎春 (1977–), 男, 讲师, 博士, 目前研究方向为非线性系统控制、动态规划, E-mail: boyingchun@sina.com;

崔黎黎 (1983–), 女, 讲师, 博士, 目前研究方向为自适应动态规划、非线性系统、微分对策、最优控制等, E-mail: cuilili8396@163.com.