

模型未知非零和博弈问题的策略迭代算法

杨明¹, 罗艳红¹, 王义贺²

(1. 东北大学信息科学与工程学院, 辽宁 沈阳 110819; 2. 国网辽宁省电力有限公司经济技术研究院, 辽宁 沈阳 110000)

摘 要: 提出了一种在线积分策略迭代算法, 用来求解内部非线性动力模型未知的双人非零和博弈问题. 通过在控制策略和干扰策略中引入探测信号, 从而避开了系统的模型信息, 得到了一个求解非零和博弈的无模型的近似动态规划算法. 该算法同步更新值函数、控制策略、扰动策略, 并且最终得到收敛的策略权值. 在算法实现过程中, 使用4个神经网络分别近似两个值函数、控制策略和扰动策略, 使用最小二乘法估计神经网络的未知参数. 最后仿真结果验证了算法的有效性.

关 键 词: 自适应动态规划; 非零和博弈; 策略迭代; 神经网络; 最优控制

中图分类号: TP 183

文献标志码: A

文章编号: 1005-3026(2015)03-0318-05

Policy Iteration Algorithm for Nonzero-Sum Games with Unknown Models

YANG Ming¹, LUO Yan-hong¹, WANG Yi-he²

(1. School of Information Science & Engineering, Northeastern University, Shenyang 110819, China;

2. Economic Technology Institute, Nation State Liaoning Province Power Co., Ltd., Shenyang 110000, China.

Corresponding author: YANG Ming, E-mail: yangming20060916@126.com)

Abstract: An online integral policy iteration algorithm was proposed to find the solution of two-player nonzero-sum differential games with completely unknown nonlinear continuous-time dynamics. Exploration signals can be added into the control and disturbance policies, rather than having to find the model information. An approximate dynamic programming (ADP) of model-free approach can be constructed, and the nonzero-sum games can be solved. The value function, control and disturbance policies simultaneously can be updated by the proposed algorithm, and converged policy weight parameters are obtained. To implement the algorithm, four neural networks are used respectively to approximate the two game value functions, the control policy and the disturbance policy. The least squares method is used to estimate the unknown parameters of the neural networks. The effectiveness of the developed scheme is demonstrated by a simulation example.

Key words: adaptive dynamic programming; nonzero-sum games; policy iteration; neural networks; optimal control

博弈论考虑游戏中的个体的预测行为和实际行为, 并研究它们的优化策略. 零和博弈表示所有博弈方的利益之和为零或一个常数, 即一方有所得, 其他方必有所失. 非零和博弈表示在不同策略组合自适应动态规划 (adaptive dynamic programming, ADP) 理论融合了动态规划、增强学习以及函数近似等方法, 在处理非线性系统的最

优控制问题中有着独特优势^[1-2]. ADP 方法已经应用到求解零和博弈问题和非零和博弈中^[3-10], 根据是否需要内部系统模型信息, 算法可分为无模型算法和有模型算法. 文献[3]提出一种基于策略迭代增强学习技术的自适应控制算法解决非零和博弈问题; 文献[7]采用积分 RL (reinforcement learning) 方法在线求解线性连续

系统的非零和博弈纳什均衡解;文献[8]采用基于单网络的ADP方法来求解非线性连续系统的非零和博弈平衡点.以上文献提出的算法均需要知道系统模型信息.文献[9]提出了一种无模型策略迭代算法求解线性二次最优问题;文献[10]提出了一种无模型策略迭代算法求解零和博弈问题,其中值函数、控制策略和干扰策略同时更新.目前还没有对内部系统模型未知的非零和博弈问题的求解.

本文提出一种在线积分策略迭代算法,该算法不需要知道系统的模型信息,利用输入输出数据求解非零和博弈问题的纳什均衡解.

1 问题描述

考虑一类非线性连续时间动态系统:

$$\dot{x} = f(x) + g(x)u + k(x)w. \quad (1)$$

其中: $x \in \mathbf{R}^n$ 是系统状态; $u \in \mathbf{R}^m$ 是控制输入; $w \in \mathbf{R}^q$ 是外部扰动输入; $f(x) \in \mathbf{R}^n$; $g(x) \in \mathbf{R}^{n \times m}$ 和 $k(x) \in \mathbf{R}^{n \times q}$ 是内部系统模型. 假设 $f(x) \in \mathbf{R}^n$, $g(x) \in \mathbf{R}^{n \times m}$ 和 $k(x) \in \mathbf{R}^{n \times q}$ 是未知的, $x=0$ 是系统(1)的一个平衡点.

仿照文献[8]定义两个性能指标:

$$\begin{aligned} J_1(x_0, u, w) &= \int_0^\infty (x^T Q_1 x + u^T R_{11} u + w^T R_{12} w) d\tau \\ &\triangleq \int_0^\infty r_1(x, u, w) d\tau; \end{aligned} \quad (2)$$

$$\begin{aligned} J_2(x_0, u, w) &= \int_0^\infty (x^T Q_2 x + u^T R_{21} u + w^T R_{22} w) d\tau \\ &\triangleq \int_0^\infty r_2(x, u, w) d\tau. \end{aligned} \quad (3)$$

其中: $Q_1 \geq 0$; $Q_2 \geq 0$; $R_{11} > 0$; $R_{12} \geq 0$; $R_{21} \geq 0$; $R_{22} > 0$.

对于控制策略 u 和干扰策略 w , 定义值函数:

$$V_1(x_i, u, w) = \int_i^\infty (x^T Q_1 x + u^T R_{11} u + w^T R_{12} w) d\tau; \quad (4)$$

$$V_2(x_i, u, w) = \int_i^\infty (x^T Q_2 x + u^T R_{21} u + w^T R_{22} w) d\tau. \quad (5)$$

定义非零和博弈问题为

$$V_1^*(x_0) = \min_u \max_d V_1(x_0, u, w), \quad (6)$$

$$V_2^*(x_0) = \min_d \max_u V_2(x_0, u, w). \quad (7)$$

问题目标是找到满足以下不等式的纳什平衡策略 (u^*, w^*) :

$$\begin{cases} V_1(x_i, u^*, w^*) \leq V_1(x_i, u, w^*), \\ V_2(x_i, u^*, w^*) \leq V_2(x_i, u^*, w). \end{cases} \quad (8)$$

对值函数微分得到非线性 Lyapunov 方程:

$$0 = r_i(x, u, w) + (\nabla V_i)^T (f(x) + g(x)u + k(x)w) \quad (i=1, 2). \quad (9)$$

其中 $\nabla V_i = \frac{\partial V_i}{\partial x}$.

定义 Hamilton 函数:

$$H_i(x, \nabla V_i, u, w) = r_i(x, u, w) + (\nabla V_i)^T (f(x) + g(x)u + k(x)w) \quad (i=1, 2).$$

由 $\frac{\partial H_i}{\partial u} = 0, \frac{\partial H_i}{\partial w} = 0$ 得到

$$\begin{cases} u = -\frac{1}{2} R_{11}^{-1} g^T(x) \nabla V_1, \\ w = -\frac{1}{2} R_{22}^{-1} k^T(x) \nabla V_2. \end{cases} \quad (10)$$

将式(10)代入式(9)得到两个耦合的 HJ 方程:

$$\begin{aligned} Q_1(x) - \frac{1}{4} (\nabla V_1)^T g(x) R_{11}^{-1} g^T(x) \nabla V_1 + (\nabla V_1)^T f(x) + \\ \frac{1}{4} (\nabla V_2)^T k(x) R_{22}^{-1} R_{12} R_{22}^{-1} k^T(x) \nabla V_2 - \\ \frac{1}{2} (\nabla V_1)^T k(x) R_{22}^{-1} k^T(x) \nabla V_2 = 0, \\ Q_2(x) - \frac{1}{4} (\nabla V_2)^T k(x) R_{22}^{-1} k^T(x) \nabla V_2 + (\nabla V_2)^T f(x) + \\ \frac{1}{4} (\nabla V_1)^T g(x) R_{11}^{-1} R_{21} R_{11}^{-1} g^T(x) \nabla V_1 - \\ \frac{1}{2} (\nabla V_2)^T g(x) R_{11}^{-1} g^T(x) \nabla V_1 = 0. \end{aligned}$$

这是两个非线性偏微分方程, 很难得到解析解.

2 无模型积分策略迭代算法

为了分析无模型积分策略迭代 (policy iteration, PI) 算法, 首先给出一个已知模型非零和博弈的 PI 算法.

算法 1 的步骤如下:

步骤 1 给定初始稳定控制策略 u_1 和干扰策略 w_1 , 设定 $i=1$.

步骤 2 (策略评价) 根据已知 u_i 和 w_i , 通过下面的李雅普诺夫方程求解 V_1^i 和 V_2^i :

$$\begin{cases} 0 = r_1(x, u_i, d_i) + (\nabla V_1^i)^T (x) (f(x) + g(x)u_i(x) + k(x)w_i(x)), \\ 0 = r_2(x, u_i, d_i) + (\nabla V_2^i)^T (x) (f(x) + g(x)u_i(x) + k(x)w_i(x)). \end{cases} \quad (11)$$

步骤 3 (策略提高) 更新控制和干扰策略:

$$\left. \begin{aligned} \mathbf{u}_{i+1}(\mathbf{x}) &= -\frac{1}{2}\mathbf{R}_{11}^{-1}\mathbf{g}^T(\mathbf{x})\nabla V_1^i(\mathbf{x}), \\ \mathbf{w}_{i+1}(\mathbf{x}) &= -\frac{1}{2}\mathbf{R}_{22}^{-1}\mathbf{k}^T(\mathbf{x})\nabla V_2^i(\mathbf{x}). \end{aligned} \right\} \quad (12)$$

步骤4 如果 $\|V_j^i - V_j^{i-1}\| \leq \varepsilon$ ($j=1,2, \varepsilon$ 是预设的正实数), 停止算法并输出 V_1^i, V_2^i ; 否则, 设 $i=i+1$, 并转到步骤2.

可以证明^[11] 当 $i \rightarrow \infty$ 时, 算法1得到的 V_j^i ($j=1,2$), $\mathbf{u}_i, \mathbf{w}_i$ 分别收敛到 HJ 方程的最优解 V_j^* , 最优纳什平衡策略 \mathbf{u}^* 和 \mathbf{w}^* .

接下来给出在线无模型积分 PI 算法. 为了规避系统模型的信息, 将探测信号 \mathbf{e} 先后加入到控制策略 \mathbf{u}_i 和干扰策略 \mathbf{w}_i 中, 假设探测信号是预知的非零有界可测信号, $\|\mathbf{e}\| \leq e_M$.

系统(1)变为

$$\left. \begin{aligned} \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})(\mathbf{u}_i + \mathbf{e}) + \mathbf{k}(\mathbf{x})\mathbf{w}_i, \\ \dot{\mathbf{x}} &= \mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{u}_i + \mathbf{k}(\mathbf{x})(\mathbf{w}_i + \mathbf{e}). \end{aligned} \right\} \quad (13)$$

值函数 V_1^i, V_2^i 关于时间的导数为

$$\left. \begin{aligned} \dot{V}_1^i &= (\nabla V_1^i)^T(\mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})(\mathbf{u}_i + \mathbf{e}) + \mathbf{k}(\mathbf{x})\mathbf{w}_i), \\ \dot{V}_2^i &= (\nabla V_2^i)^T(\mathbf{f}(\mathbf{x}) + \mathbf{g}(\mathbf{x})\mathbf{u}_i + \mathbf{k}(\mathbf{x})(\mathbf{w}_i + \mathbf{e})). \end{aligned} \right\} \quad (14)$$

利用式(11), 式(12)从 t 到 $t+T$ 积分式(14)得到

$$\begin{aligned} V_1^i(\mathbf{x}_{t+T}) - V_1^i(\mathbf{x}_t) &= -\int_t^{t+T} r_1(\mathbf{x}, \mathbf{u}_i, \mathbf{w}_i) d\tau - \\ &\quad 2\int_t^{t+T} \mathbf{u}_{i+1}^T \mathbf{R}_{11} \mathbf{e} d\tau, \end{aligned} \quad (15)$$

$$\begin{aligned} V_2^i(\mathbf{x}_{t+T}) - V_2^i(\mathbf{x}_t) &= -\int_t^{t+T} r_2(\mathbf{x}, \mathbf{u}_i, \mathbf{w}_i) d\tau - \\ &\quad 2\int_t^{t+T} \mathbf{w}_{i+1}^T \mathbf{R}_{22} \mathbf{e} d\tau. \end{aligned} \quad (16)$$

其中 $\mathbf{x}(t)$ 和 $\mathbf{x}(t+T)$ 记为 \mathbf{x}_t 和 \mathbf{x}_{t+T} .

在式(15)和式(16)中没有出现 $\mathbf{f}(\mathbf{x}), \mathbf{g}(\mathbf{x})$ 和 $\mathbf{k}(\mathbf{x})$ 的信息, 因此得到了下面的无模型积分 PI 算法.

算法2的步骤如下:

步骤1 给定初始稳定控制策略 \mathbf{u}_1 和干扰策略 \mathbf{w}_1 , 设定 $i=1$.

步骤2 已知 $\mathbf{u}_i, \mathbf{w}_i, \mathbf{e}$ 解下面的方程求解 $V_1^i, V_2^i, \mathbf{u}_{i+1}, \mathbf{w}_{i+1}$,

$$\begin{aligned} V_1^i(\mathbf{x}_t) &= V_1^i(\mathbf{x}_{t+T}) + \int_t^{t+T} r_1(\mathbf{x}, \mathbf{u}_i, \mathbf{w}_i) + \\ &\quad 2\int_t^{t+T} \mathbf{u}_{i+1}^T \mathbf{R}_{11} \mathbf{e} d\tau, \end{aligned} \quad (17)$$

$$\begin{aligned} V_2^i(\mathbf{x}_t) &= V_2^i(\mathbf{x}_{t+T}) + \int_t^{t+T} r_2(\mathbf{x}, \mathbf{u}_i, \mathbf{w}_i) + \\ &\quad 2\int_t^{t+T} \mathbf{w}_{i+1}^T \mathbf{R}_{22} \mathbf{e} d\tau. \end{aligned} \quad (18)$$

步骤3 如果 $\|V_j^i - V_j^{i-1}\| \leq \varepsilon$ ($j=1,2, \varepsilon$ 是预设的正实数), 停止算法并输出 V_1^i, V_2^i ; 否则, 设

$i=i+1$, 并转到步骤2.

在算法2中同时更新值函数、控制策略和干扰策略.

3 算法实现

为了实现算法2, 使用2个评价神经网络和2个执行网络用来近似值函数、控制策略和干扰策略. 这里为了简化表示, 记 $m=q=1$. 对式(17)和式(18), $V_j^i(\mathbf{x})$ ($j=1,2$), $\mathbf{u}_{i+1}(\mathbf{x}), \mathbf{w}_{i+1}(\mathbf{x})$ 可以用单层神经网络表示如下:

$$V_1^i(\mathbf{x}) = (\mathbf{W}_1^i)^T \boldsymbol{\phi}_1(\mathbf{x}) + \varepsilon_1^i(\mathbf{x}); \quad (19)$$

$$V_2^i(\mathbf{x}) = (\mathbf{W}_2^i)^T \boldsymbol{\phi}_2(\mathbf{x}) + \varepsilon_2^i(\mathbf{x}); \quad (20)$$

$$\mathbf{u}_{i+1}(\mathbf{x}) = (\mathbf{W}_{i+1}^3)^T \boldsymbol{\phi}_3(\mathbf{x}) + \varepsilon_{i+1}^3(\mathbf{x}); \quad (21)$$

$$\mathbf{w}_{i+1}(\mathbf{x}) = (\mathbf{W}_{i+1}^4)^T \boldsymbol{\phi}_4(\mathbf{x}) + \varepsilon_{i+1}^4(\mathbf{x}). \quad (22)$$

其中: \mathbf{W}_i^j ($j=1,2,3,4$) 是具有适当维数的未知理想权值; $\boldsymbol{\phi}_j$ ($j=1,2,3,4$) 是激活函数, 采用多项式函数; ε_i^j ($j=1,2,3,4$) 是有界神经网络近似误差. 当隐藏层的神经元数量趋于无穷时, 近似误差一致趋于零. 因为理想的权值未知, 假设评价网络和执行网络的输出为

$$\hat{V}_1^i(\mathbf{x}) = (\hat{\mathbf{W}}_1^i)^T \boldsymbol{\phi}_1(\mathbf{x}); \quad (23)$$

$$\hat{V}_2^i(\mathbf{x}) = (\hat{\mathbf{W}}_2^i)^T \boldsymbol{\phi}_2(\mathbf{x}); \quad (24)$$

$$\hat{\mathbf{u}}_{i+1}(\mathbf{x}) = (\hat{\mathbf{W}}_{i+1}^3)^T \boldsymbol{\phi}_3(\mathbf{x}); \quad (25)$$

$$\hat{\mathbf{w}}_{i+1}(\mathbf{x}) = (\hat{\mathbf{W}}_{i+1}^4)^T \boldsymbol{\phi}_4(\mathbf{x}). \quad (26)$$

其中 $\hat{\mathbf{W}}_i^1, \hat{\mathbf{W}}_i^2, \hat{\mathbf{W}}_{i+1}^3, \hat{\mathbf{W}}_{i+1}^4$ 是当前估计值.

使用式(23)~式(26), 式(17)和式(18)可以写成下面的形式:

$$(\boldsymbol{\psi}_k^1)^T \begin{bmatrix} \hat{\mathbf{W}}_1^1 \\ \hat{\mathbf{W}}_{i+1}^3 \end{bmatrix} = \theta_k^1; \quad (27)$$

$$(\boldsymbol{\psi}_k^2)^T \begin{bmatrix} \hat{\mathbf{W}}_2^2 \\ \hat{\mathbf{W}}_{i+1}^4 \end{bmatrix} = \theta_k^2. \quad (28)$$

其中: $\theta_k^j = \int_{t+(k-1)T}^{t+kT} r_j(\mathbf{x}, \mathbf{u}_i, \mathbf{w}_i) d\tau$ ($j=1,2$);

$$\boldsymbol{\psi}_k^1 = [(\boldsymbol{\phi}_1(\mathbf{x}_{t+(k-1)T}) - \boldsymbol{\phi}_1(\mathbf{x}_{t+kT}))^T, -2\int_{t+(k-1)T}^{t+kT} \mathbf{R}_{11} \mathbf{e} \boldsymbol{\phi}_3^T(\mathbf{x}) d\tau];$$

$$\boldsymbol{\psi}_k^2 = [(\boldsymbol{\phi}_2(\mathbf{x}_{t+(k-1)T}) - \boldsymbol{\phi}_2(\mathbf{x}_{t+kT}))^T, -2\int_{t+(k-1)T}^{t+kT} \mathbf{R}_{22} \mathbf{e} \boldsymbol{\phi}_4^T(\mathbf{x}) d\tau].$$

因为式(27)和式(28)是一维方程, 不能保证解的唯一性. 使用最小二乘法计算参数向量:

$$(\boldsymbol{\Phi}_i^1)^T \begin{bmatrix} \hat{\mathbf{W}}_1^1 \\ \hat{\mathbf{W}}_{i+1}^3 \end{bmatrix} = \boldsymbol{\Theta}_i^1, (\boldsymbol{\Phi}_i^2)^T \begin{bmatrix} \hat{\mathbf{W}}_2^2 \\ \hat{\mathbf{W}}_{i+1}^4 \end{bmatrix} = \boldsymbol{\Theta}_i^2.$$

其中: $\boldsymbol{\Phi}_i^j = [\boldsymbol{\psi}_1^j, \dots, \boldsymbol{\psi}_K^j]$ ($j=1,2$);

$$\boldsymbol{\Theta}_i^j = [\theta_1^j, \dots, \theta_K^j]$$
 ($j=1,2$).

如果 Φ_i^j 列满秩, 则:

$$\begin{bmatrix} \hat{W}_i^1 \\ \hat{W}_{i+1}^3 \end{bmatrix} = (\Phi_i^1 (\Phi_i^1)^T)^{-1} \Phi_i^1 \Theta_i^1;$$
$$\begin{bmatrix} \hat{W}_i^2 \\ \hat{W}_{i+1}^4 \end{bmatrix} = (\Phi_i^2 (\Phi_i^2)^T)^{-1} \Phi_i^2 \Theta_i^2.$$

因此采样数据 K 要取得足够大, $K_{\min} = \text{rank}(\Phi_i^j) = N_1 + N_2$ (N_1, N_2 分别对应 \hat{W}_i^1 和 \hat{W}_{i+1}^3 的维数), 以保证 $(\Phi_i^1 (\Phi_i^1)^T)^{-1}$ 存在. 最小二乘问题可以通过采集系统数据实时求解.

4 仿真例子

为了验证算法的有效性, 本文利用 Matlab 软件对算法 2 进行了仿真实验.

考虑下面的非线性系统:

$\dot{x} = f(x) + g(x)u + k(x)w.$

其中:

$f(x) = \begin{bmatrix} x_2 - 2x_1 \\ -x_2 - 0.5x_1 + 0.25x_2(\cos(2x_1 + 2))^2 + \\ 0.25x_2(\sin(4x_1^2) + 2)^2 \end{bmatrix};$

$g(x) = \begin{bmatrix} 0 \\ \cos(2x_1 + 2) \end{bmatrix}; k(x) = \begin{bmatrix} 0 \\ \sin(4x_1^2) + 2 \end{bmatrix}.$

性能指标 (2), (3) 分别定义为 $Q_1(x) = 2x^T x, R_{11} = R_{12} = 2I, Q_2(x) = x^T x, R_{21} = R_{22} = I$, 其中 I 代表单位矩阵.

设置初始状态 $x_0 = [1 \ 2]^T$. 神经网络激活函数 $\phi_1(x) = \phi_2(x) = \phi_3(x) = \phi_4(x) = [x_1^2 \ x_1 x_2 \ x_2^2]^T$, 其初始参数均设置为零. 探测信号选择 $e = \sin(t)$, 算法在线执行, 系统响应数据采集周期 $T = 0.04$, 最小二乘法需要 20 组数据, 所以每 0.8 s 更新一次激活函数参数. 图 1 表示 V_1 的

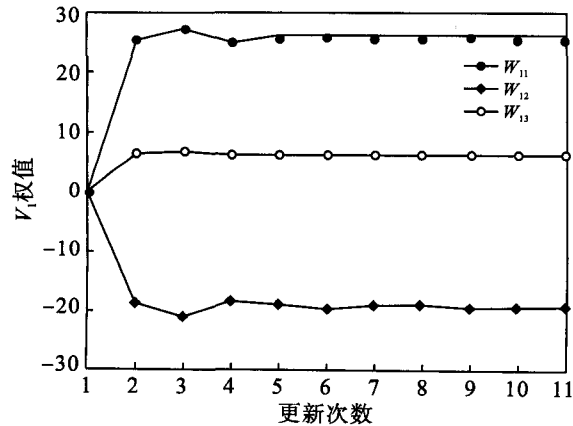


图 1 V_1 参数更新
Fig. 1 V_1 parameter update

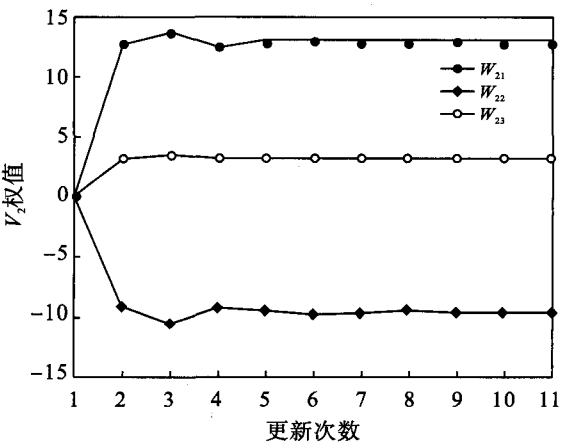


图 2 V_2 参数更新
Fig. 2 V_2 parameter update

参数对更新次数的变化, 图 2 表示 V_2 的参数对更新次数的变化, 可以看出在 3 次更新以后参数基本保持不变. 图 3 表示在得到控制器和干扰器作用下的系统响应图.

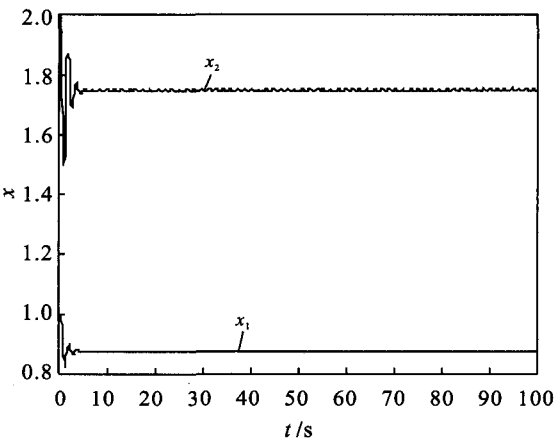


图 3 系统状态响应
Fig. 3 System state response

从图 1 和图 2 可以看出, 值函数的权重 W 很快就收敛到了稳定值. 从图 3 可以看出得到的控制器可以保证系统一致最终有界的.

5 结 语

本文提出一种无模型积分策略迭代算法求解非零和博弈问题, 通过在控制策略和干扰策略中加入已知噪声, 消除算法迭代对系统信息模型的依赖. 值函数、控制策略和干扰策略同时更新, 加快了算法收敛速度. 最后仿真算例验证了算法的有效性.

通过图 1 和图 2 可以看出,文献[5]中所得到的闭环系统的零输入响应抖动较大,收敛速度较慢;本文所得到的闭环系统的零输入响应抖动较小,而且收敛速度相对较快,具有更好的鲁棒性。

5 结 语

本文针对 NGHS 算法因位置更新操作导致过早收敛,容易陷入局部最优的问题,给出了新的位置更新策略,提出的 IGHS 算法增强了算法开发解空间局部信息的能力,避免了算法因收敛过快而陷入局部最优的不足。

将所提出算法应用于线性系统鲁棒极点配置问题中,解决了以往条件数优化计算中需要凸转化处理的不足。仿真实验表明,利用本文方法所得到的条件数明显优于 Method1, Newton 和 LMI 方法。从条件数值和反馈增益矩阵的范数来看,本文方法也略优于 RNN 方法,同时反馈后的闭环系统具有更好的鲁棒性。

参考文献:

- [1] Kautsky J, Nichols N K, Van Dooren P. Robust pole assignment in linear state feedback[J]. *International Journal of Control*, 1985, 41(5): 1129 - 1155.
- [2] Byers R, Nash S G. Approaches to robust pole assignment

[J]. *International Journal of Control*, 1989, 49(1): 97 - 117.

- [3] Rami A M, El Faiz S, Benzaouia A, et al. Robust exact pole placement via an LMI-based algorithm [J]. *IEEE Transactions on Automatic Control*, 2009, 54(2): 394 - 398.
- [4] Le X, Wang J. Robust pole assignment for synthesizing feedback control systems using recurrent neural networks[J]. *IEEE Transactions on Neural Networks and Learning Systems*, 2014, 25(2): 383 - 393.
- [5] Geem Z W, Kim J H, Loganathan G V. A new heuristic optimization algorithm; harmony search [J]. *Simulation*, 2001, 76(2): 60 - 68.
- [6] Pan Q K, Suganthan P N, Tasgetiren M F, et al. A self-adaptive global best harmony search algorithm for continuous optimization problems [J]. *Applied Mathematics and Computation*, 2010, 216(3): 830 - 848.
- [7] Zou D, Gao L, Wu J, et al. Novel global harmony search algorithm for unconstrained problems[J]. *Neurocomputing*, 2010, 73(16): 3308 - 3318.
- [8] Shreem S S, Abdullah S, Nazri M Z A. Hybridising harmony search with a Markov blanket for gene selection problems [J]. *Information Sciences*, 2014, 258: 108 - 121.
- [9] Jeddi B, Vahidinasab V. A modified harmony search method for environmental/economic load dispatch of real-world power systems [J]. *Energy Conversion and Management*, 2014, 78: 661 - 675.
- [10] Jiang D, Wang J. Augmented gradient flows for on-line robust pole assignment via state and output feedback [J]. *Automatica*, 2002, 38(2): 279 - 286.

(上接第 321 页)

参考文献:

- [1] 张化光, 张欣, 罗艳红, 等. 自适应动态规划综述[J]. *自动化学报*, 2013, 39(4): 303 - 311.
(Zhang Hua-guang, Zhang Xin, Luo Yan-hong, et al. An overview of research on adaptive dynamic programming[J]. *ACTA Automatica Sinica*, 2013, 39(4): 303 - 311.)
- [2] 刘德荣, 李宏亮, 王鼎. 基于数据的自学习优化控制: 研究进展与展望[J]. *自动化学报*, 2013, 39(11): 1858 - 1870.
(Liu De-rong, Li Hong-liang, Wang Ding. Data-based self-learning optimal control; research progress and prospects[J]. *ACTA Automatica Sinica*, 2013, 39(11): 1858 - 1870.)
- [3] Vamvoudakis K G, Lewis F L. Multi-player non-zero-sum games; online adaptive learning solution of coupled Hamilton-Jacobi equations[J]. *Automatica*, 2011, 47(8): 1556 - 1569.
- [4] Abu-Khalaf M, Lewis F L, Jie H. Neurodynamic programming and zero-sum games for constrained control systems[J]. *IEEE Transactions on Neural Networks*, 2008, 19(7): 1243 - 1252.
- [5] Al-Tamimi A, Abu-Khalaf M, Lewis F L. Adaptive critic designs for discrete-time zero-sum games with application to H infinity control[J]. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 2007, 37(1): 240 - 247.

- [6] Zhang H, Wei Q, Liu D. An iterative adaptive dynamic programming method for solving a class of nonlinear zero-sum differential games[J]. *Automatica*, 2011, 47(1): 207 - 214.
- [7] Vrabie D, Lewis F. Integral reinforcement learning for online computation of feedback Nash strategies of nonzero-sum differential games [C]// 2010 49th IEEE Conference on Decision and Control (CDC). Atlanta, 2010: 3066 - 3071.
- [8] Zhang H G, Cui L L, Luo Y H. Near-optimal control for nonzero-sum differential games of continuous-time nonlinear systems using single-network ADP[J]. *IEEE Transactions on Cybernetics*, 2013, 43(1): 206 - 216.
- [9] Jiang Y, Jiang Z P. Computational adaptive optimal control for continuous-time linear systems with completely unknown dynamics[J]. *Automatica*, 2012, 48(10): 2699 - 2704.
- [10] Li H, Liu D, Wang D. Integral policy iteration for zero-sum games with completely unknown nonlinear dynamics [C]// Neural Information Processing, 20th International Conference, ICONIP 2013. Berlin Heidelberg: Springer, 2013: 225 - 232.
- [11] Gajic Z, Li T Y. Simulation results for two new algorithms for solving coupled algebraic Riccati equations [C]// The Third International Symposium on Differential Games. Nice, 1988.