Journal of Zhejiang University-SCIENCE C (Computers & Electronics)
ISSN 1869-1951 (Print); ISSN 1869-196X (Online)
www.zju.edu.cn/jzus; www.springerlink.com
E-mail: jzus@zju.edu.cn

Adaptive dynamic programming for linear impulse systems*

Xiao-hua WANG†1,2, Juan-juan YU1, Yao HUANG1, Hua WANG1,2, Zhong-hua MIAO†‡1,2

(*1School of Mechatronics Engineering and Automation, Shanghai University, Shanghai 200072, China*)

(*2Shanghai Key Laboratory of Power Station Automation Technology, Shanghai University, Shanghai 200072, China*)

†E-mail: {x.wang, zhhmiao}@shu.edu.cn

Received May 27, 2013; Revision accepted Nov. 26, 2013; Crosschecked Dec. 19, 2013

Abstract: We investigate the optimization of linear impulse systems with the reinforcement learning based adaptive dynamic programming (ADP) method. For linear impulse systems, the optimal objective function is shown to be a quadric form of the pre-impulse states. The ADP method provides solutions that iteratively converge to the optimal objective function. If an initial guess of the pre-impulse objective function is selected as a quadratic form of the pre-impulse states, the objective function iteratively converges to the optimal one through ADP. Though direct use of the quadratic objective function of the states within the ADP method is theoretically possible, the numerical singularity problem may occur due to the matrix inversion therein when the system dimensionality increases. A neural network based ADP method can circumvent this problem. A neural network with polynomial activation functions is selected to approximate the pre-impulse objective function and trained iteratively using the ADP method to achieve optimal control. After a successful training, optimal impulse control can be derived. Simulations are presented for illustrative purposes.

Key words: Adaptive dynamic programming (ADP), Impulse system, Optimal control, Neural network
doi:10.1631/jzus.C1300145 **Document code:** A **CLC number:** TP273.1

1 Introduction

Impulse system control has attracted much attention recently (Lakshmikantham *et al.*, 1989; Bainov and Simeonov, 1995). An impulsive differential equation (Lakshmikantham *et al.*, 1989) provides a fundamental tool for impulse system modeling and control. When the time of impulse is fixed, the impulse system is known as a fixed time impulse system; when the impulse time is a function of system states, it is a variable impulse system problem (Wang, 2008). An interesting and ubiquitous example of an impulse system is that of human beings (plant) taking medicine such as tablets (impulse control). Yang (1999) gave a few other good examples.

Optimal control of impulse systems has been studied recently. The existence of optimal control has been investigated (Ahmed, 2003; Wang and Yang, 2010). Necessary conditions for optimality have been proposed for different classes of impulse systems (Silva and Vinter, 1997; Liu *et al.*, 2008). Methods of dynamic programming (Kurzhanski and Daryin, 2008) and maximum principle (Wu and Zhang, 2011; Fraga and Pereira, 2012) have been studied in the literature. However, to numerically solve for the optimal impulse control is still a major challenge.

Adaptive dynamic programming (ADP) is a reinforcement learning based method. It was first proposed by Werbos (1974) to solve for the optimal

‡ Corresponding author

* Project supported by the National Natural Science Foundation of China (Nos. 61104006, 51175319, and 11202121), the MOE Scientific Research Foundation for the Returned Overseas Chinese Scholars, the Natural Science Foundation of Shanghai (No. 11ZR1412400), and the Shanghai Education Commission (Nos. 12YZ010, 12JC1404100, and 11CH-05), China

control forward in time and further discussed in Werbos *et al.* (1992) and Werbos (2008), and has been widely used in optimization of both continuous and discrete systems (Balakrishnan *et al.*, 2008; Lewis and Vrabie, 2009; Wang *et al.*, 2009). Recently, this method has been rapidly developing (Bertsekas, 2011). Together with the adaptive method, ADP can handle optimality under system parameter uncertainty (Dierks and Jagannathan, 2011; Jiang and Jiang, 2012; 2013). Its online implementation has also been investigated (Vamvoudakis and Lewis, 2010).

Since the ADP method originates from Bellman's principle of optimality, the fundamental idea is also applicable to impulse systems. Wang *et al.* (2012) gave some results on optimization of a linear, fixed time impulse system using ADP, where the optimal objective function results in a quadratic function of the states. When the system dimensionality increases, singularity problems may occur due to matrix inversion. Therefore, an improved ADP algorithm is proposed in this paper. A neural network is used to approximate the pre-impulse objective function, whose weights are iteratively trained using the ADP method. After the training is complete, the network outputs the optimal objective function value and the optimal impulse control can be derived accordingly.

2 System model and optimization problem statement

2.1 System model

Consider the following fixed-time linear impulse system model:

$$\begin{cases} \dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x}, & t \neq \tau_i, \\ \boldsymbol{x}_i^+ = \boldsymbol{x}_i^- + \boldsymbol{B}\boldsymbol{u}_i, & t = \tau_i, \end{cases} \tag{1}$$

where $\boldsymbol{A} \in \mathbb{R}^{n\times n}$, $\boldsymbol{B} \in \mathbb{R}^{n\times m}$, $i \in \mathbb{Z}^+$, $\boldsymbol{x}_0$ is the initial state, τ_i is the known impulse moment, superscripts '+' and '−' denote the right and left limits with respect to the impulse moments, respectively, $i = 1, 2, \cdots$ is the index of the moments when impulses occur, and τ_i^- and τ_i^+ are referred to as pre- and post-impulse moments, respectively. In this study, the time between two consecutive impulses is considered to be fixed, i.e., $\tau_i - \tau_{i-1} = \delta\tau = \text{const.}$

Fig. 1 describes the system dynamics along the time axis. Note that between two consecutive impulses, for example, τ_i^+ and τ_{i+1}^- , continuous dynamics are dominant. At the time when an impulse is given, say between τ_i^- and τ_i^+, the impulsive dynamics are dominant.

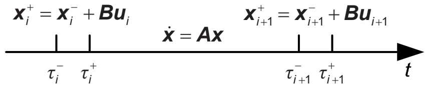


Fig. 1 System dynamics along the time line

For brevity, $\boldsymbol{u}_i$ denotes $\boldsymbol{u}(\tau_i)$, $\boldsymbol{x}_i^-$ denotes $\boldsymbol{x}(\tau_i^-)$, and $\boldsymbol{x}_i^+$ denotes $\boldsymbol{x}(\tau_i^+)$. A variable at τ_i^- or τ_i^+ is abbreviated to a variable with subscript 'i' and superscript '−' or '+', respectively.

2.2 Objective function

Consider the quadratic objective function given below:

$$J = \sum_{i=1}^{\infty}\left(\frac{1}{2}\boldsymbol{u}_i^{\mathrm{T}}\boldsymbol{R}\boldsymbol{u}_i + \int_{\tau_i^+}^{\tau_{i+1}^-}\frac{1}{2}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{Q}\boldsymbol{x}\mathrm{d}t\right), \tag{2}$$

where $\boldsymbol{R} \in \mathbb{R}^{m\times m} > \boldsymbol{0}$ and $\boldsymbol{Q} \in \mathbb{R}^{n\times n} \geq \boldsymbol{0}$.

The problem studied here is an infinite horizon optimal impulse control problem and the aim is to find the optimal impulse control of system (1) while minimizing the objective function (2).

Since $\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x}$ between two consecutive impulses, states between impulses are given by $\boldsymbol{x}(t) = \mathrm{e}^{\boldsymbol{A}(t-\tau_i^+)}\boldsymbol{x}_i^+$ for $t \in \left[\tau_i^+, \tau_{i+1}^-\right]$. Therefore, $\boldsymbol{x}_{i+1}^- = \mathrm{e}^{\boldsymbol{A}(\tau_{i+1}^- - \tau_i^+)}\boldsymbol{x}_i^+$. For $\tau_{i+1} - \tau_i = \delta\tau = \text{const.}$, by defining

$$\boldsymbol{\Psi} \triangleq \mathrm{e}^{\boldsymbol{A}\delta\tau}, \tag{3}$$

we have

$$\boldsymbol{x}_{i+1}^- = \boldsymbol{\Psi}\boldsymbol{x}_i^+ = \boldsymbol{\Psi}\left(\boldsymbol{x}_i^- + \boldsymbol{B}\boldsymbol{u}_i\right). \tag{4}$$

The $\int_{\tau_i^+}^{\tau_{i+1}^-}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{Q}\boldsymbol{x}\mathrm{d}t$ term, part of utility function (2), can then be rearranged as follows:

$$\begin{aligned} &\int_{\tau_i^+}^{\tau_{i+1}^-}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{Q}\boldsymbol{x}\mathrm{d}t \\ =&\int_{\tau_i^+}^{\tau_{i+1}^-}\left(\mathrm{e}^{\boldsymbol{A}^{\mathrm{T}}(t-\tau_i^+)}\boldsymbol{x}_i^+\right)^{\mathrm{T}}\boldsymbol{Q}\left(\mathrm{e}^{\boldsymbol{A}(t-\tau_i^+)}\boldsymbol{x}_i^+\right)\mathrm{d}t \\ =&\left(\boldsymbol{x}_i^+\right)^{\mathrm{T}}\int_{\tau_i^+}^{\tau_{i+1}^-}\mathrm{e}^{\boldsymbol{A}^{\mathrm{T}}(t-\tau_i^+)}\boldsymbol{Q}\mathrm{e}^{\boldsymbol{A}(t-\tau_i^+)}\mathrm{d}t\left(\boldsymbol{x}_i^+\right) \\ =&(\boldsymbol{x}_i^+)^{\mathrm{T}}\int_0^{\delta\tau}\mathrm{e}^{\boldsymbol{A}^{\mathrm{T}}\tau}\boldsymbol{Q}\mathrm{e}^{\boldsymbol{A}\tau}\mathrm{d}\tau(\boldsymbol{x}_i^+) \\ =&(\boldsymbol{x}_i^+)^{\mathrm{T}}\bar{\boldsymbol{Q}}(\boldsymbol{x}_i^+), \end{aligned} \tag{5}$$

where

$$\bar{\boldsymbol{Q}} \triangleq \int_0^{\delta\tau} \mathrm{e}^{\boldsymbol{A}^{\mathrm{T}}\tau}\boldsymbol{Q}\mathrm{e}^{\boldsymbol{A}\tau}\mathrm{d}\tau. \tag{6}$$

3 System properties

3.1 Lemmas and theorems

Definition 1 (Impulsive controllability) (Liu, 1995) A system is said to be impulsively controllable if for vectors $\boldsymbol{x}_0$ and $\boldsymbol{x}_f \in \mathbb{R}^n$, and any open interval (t_0, t_f), there exist real numbers $\tau_i \in [t_0, t_f]$, $\tau_1 < \tau_2 < \cdots < \tau_k$, and vectors $\boldsymbol{u}_i \in \mathbb{R}^{m\times 1}$, $i = 1, 2, \cdots, k < \infty$ such that system (1) has a solution $\boldsymbol{x}(t)$ within (t_0, t_f) satisfying $\boldsymbol{x}(t_0) = \boldsymbol{x}_0$ and $\boldsymbol{x}(t_f^+) = \boldsymbol{x}_f$.

Lemma 1 (Impulsive controllability condition) (Liu, 1995) Impulse system (1) is impulsively controllable if and only if

$$\mathrm{Rank}[\boldsymbol{B}, \boldsymbol{AB}, \boldsymbol{A}^2\boldsymbol{B}, \cdots, \boldsymbol{A}^{n-1}\boldsymbol{B}] = n. \tag{7}$$

Theorem 1 (Existence of optimal impulse feedback control) (Wang and Balakrishnan, 2010; Wang *et al.*, 2012) Assuming system (1) is impulsively controllable, $\boldsymbol{Q} = \boldsymbol{C}^{\mathrm{T}}\boldsymbol{C}$, and $[\boldsymbol{C}^{\mathrm{T}}, \boldsymbol{A}^{\mathrm{T}}\boldsymbol{C}^{\mathrm{T}}, \cdots, (\boldsymbol{A}^{n-1})^{\mathrm{T}}\boldsymbol{C}^{\mathrm{T}}]^{\mathrm{T}}$ has full rank, the optimal impulse control of system (1) with respect to objective function (2) can be written as

$$\boldsymbol{u}_i = -\boldsymbol{R}^{-1}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{P}^-\boldsymbol{x}_i^-, \tag{8}$$

where $\boldsymbol{P}$ satisfies the following equations:

$$\dot{\boldsymbol{P}} = -\boldsymbol{A}^{\mathrm{T}}\boldsymbol{P} - \boldsymbol{PA} - \boldsymbol{Q}, \quad t \neq \tau_i, \tag{9}$$

$$\boldsymbol{P}^- = \boldsymbol{P}^+(\boldsymbol{I} + \boldsymbol{BR}^{-1}\boldsymbol{B}^{\mathrm{T}}\boldsymbol{P}^+)^{-1}, \quad t = \tau_i. \tag{10}$$

The dynamic equation (9) defines the relationship between $\boldsymbol{P}_i^+$ and $\boldsymbol{P}_{i+1}^-$. Together with the propagation equation (10) between $\boldsymbol{P}_i^+$ and $\boldsymbol{P}_i^-$, we can obtain the explicit steady state expression of pre-impulse $\boldsymbol{P}^-$, which is

$$\boldsymbol{P}^- = \left[(\boldsymbol{\varPsi}^{\mathrm{T}}\boldsymbol{P}^-\boldsymbol{\varPsi} + \bar{\boldsymbol{Q}})^{-1} + \boldsymbol{BR}^{-1}\boldsymbol{B}^{\mathrm{T}}\right]^{-1}. \tag{11}$$

With the optimal impulse control (8), the impulse system (1) is asymptotically stabilized. The pre-impulse optimal objective function is equal to

$$J(\tau_i^-)^* = \frac{1}{2}(\boldsymbol{x}_i^-)^{\mathrm{T}}\boldsymbol{P}^-\boldsymbol{x}_i^-. \tag{12}$$

Theorem 2 (Impulsive ADP method and its convergence) (Wang *et al.*, 2012) Assuming system (1) is impulsively controllable and the optimal impulse control exists, we begin with an initial pre-impulse objective function $V^0(\boldsymbol{x}_i^-) = \dfrac{1}{2}(\boldsymbol{x}_i^-)^{\mathrm{T}}\boldsymbol{P}_0^-\boldsymbol{x}_i^-$, where $\boldsymbol{P}_0^- \geq \boldsymbol{0}$. Update impulse control $\boldsymbol{u}_i$ at moment τ_i iteratively according to the following equation:

$$\begin{aligned}\boldsymbol{u}_i^k =& \arg\min_{\boldsymbol{u}_i^k}[\frac{1}{2}(\boldsymbol{u}_i^k)^{\mathrm{T}}\boldsymbol{R}\boldsymbol{u}_i^k + \frac{1}{2}\int_{\tau_i^+}^{\tau_{i+1}^-}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{Q}\boldsymbol{x}\mathrm{d}t \\ &+ V^k(\boldsymbol{x}_{i+1}^-)],\end{aligned} \tag{13}$$

where k is the iteration index and $k = 0, 1, \cdots$.

After $\boldsymbol{u}_k$ is updated successfully using Eq. (13), update the pre-impulse value function $V^k(\boldsymbol{x}) \geq 0$ as follows:

$$\begin{aligned}V^{k+1}(\boldsymbol{x}_i^-) &= \frac{1}{2}(\boldsymbol{x}_i^-)^{\mathrm{T}}\boldsymbol{P}_{k+1}^-\boldsymbol{x}_i^- \\ &= \min[\frac{1}{2}\boldsymbol{u}_i{}^{\mathrm{T}}\boldsymbol{R}\boldsymbol{u}_i + \frac{1}{2}\int_{\tau_i^+}^{\tau_{i+1}^-}\boldsymbol{x}^{\mathrm{T}}\boldsymbol{Q}\boldsymbol{x}\mathrm{d}t + V^k(\boldsymbol{x}_{i+1}^-)].\end{aligned} \tag{14}$$

Repeat the above process as shown in Eqs. (13) and (14). The objective function sequence V^k iteratively converges to the optimal V^*, and $\boldsymbol{u}_i^k$ converges to the optimal impulse $\boldsymbol{u}_i^*$. Note that the superscript '$*$' denotes the optimal values.

3.2 Remarks

Theorem 1 provides an explicit expression of the optimal impulse control in the feedback form, i.e., Eq. (8). The $\boldsymbol{P}$ dynamics are hybrid, composed of an update equation (10) at impulse moment τ_i and a propagation differential equation (9) between two consecutive impulse moments. It is difficult to directly calculate the optimal impulse control using the explicit expression of $\boldsymbol{P}^-$ in Eq. (11).

Theorem 2 presents the ADP method that enables optimal impulse control. The ADP method starts with a semi-positive $\boldsymbol{P}_0^-$ and assumes an objective function $(\boldsymbol{x}_i^-)^{\mathrm{T}}\boldsymbol{P}_k^-\boldsymbol{x}_i^-$. Through the iterations, $\boldsymbol{P}_k^-$ converges to the optimal $\boldsymbol{P}^-$ and the pre-impulse objective function $V^k = \dfrac{1}{2}(\boldsymbol{x}_i^-)^{\mathrm{T}}\boldsymbol{P}_k^-\boldsymbol{x}_i^-$ tends to the optimal one in Eq. (12). As the objective function follows a quadratic form of the pre-impulse states $\boldsymbol{x}_i^-$, Eqs. (13) and (14) actually contain a large amount of matrix inversions, which leads to numerical difficulties when the system dimensionality increases. Though the convergence proof is mathematically rigorous, the iterative process may diverge numerically.

4 Solutions for the optimal impulse control using ADP with neural networks

If iterative training of the objective function is carried out using the direct form of $V^k(\boldsymbol{x}_i^-) = \frac{1}{2}(\boldsymbol{x}_i^-)^{\mathrm{T}}\boldsymbol{P}_k^-\boldsymbol{x}_i^-$, the ADP algorithm becomes an iterative process as follows:

$$\boldsymbol{P}_{k+1}^- = \left[\left(\boldsymbol{\varPsi}^{\mathrm{T}}\boldsymbol{P}_k{}^-\boldsymbol{\varPsi} + \bar{\boldsymbol{Q}}\right)^{-1} + \boldsymbol{B}\boldsymbol{R}^{-1}\boldsymbol{B}^{\mathrm{T}}\right]^{-1}, \quad (15)$$

where $\boldsymbol{\varPsi}$ and $\bar{\boldsymbol{Q}}$ are as defined in Eqs. (3) and (6), respectively. The convergence of the $\boldsymbol{P}_k^-$ sequence to the optimal $\boldsymbol{P}^-$ has been shown in Wang *et al.* (2012).

When the system dimensionality increases, singularity problems may occur due to matrix inversions during the iterative calculation. To deal with this issue, a neural network based ADP method is proposed in this section.

4.1 Neural network approximation

It is well known that neural networks can be used to approximate smooth functions. In this study, we select a neural network with polynomial activation functions, whose weights are iteratively trained to approximate the optimal objective function. After a successful approximation, the optimal impulse control can be derived accordingly.

The objective function using the network approximation is written as follows:

$$V^k(\boldsymbol{x}_i^-) = \boldsymbol{W}^{\mathrm{T}}\boldsymbol{\varPhi}(\boldsymbol{x}_i^-) = \sum_{j=1}^{L}\boldsymbol{W}_j^k\boldsymbol{\varPhi}_j(\boldsymbol{x}_i^-), \quad (16)$$

where L is the number of activation functions, $\boldsymbol{\varPhi}(\boldsymbol{x}_i^-) = [\varPhi_1(\boldsymbol{x}_i^-), \varPhi_2(\boldsymbol{x}_i^-), \cdots, \varPhi_L(\boldsymbol{x}_i^-)]^{\mathrm{T}}$ is the activation function vector, and $\boldsymbol{W} = [W_1, W_2, \cdots, W_L]^{\mathrm{T}}$ is the weight vector. k is the iteration index. The weight vector is initialized as a zero vector, i.e.,

$$\boldsymbol{W}^0 = \boldsymbol{0}_{L\times 1}. \quad (17)$$

Using network approximation, Eqs. (13) and (14) become

$$\boldsymbol{R}\boldsymbol{u}_i^k + (\frac{\partial \boldsymbol{x}_i^+}{\partial \boldsymbol{u}_i^k})^{\mathrm{T}}\bar{\boldsymbol{Q}}\boldsymbol{x}_i^+ + (\frac{\partial \boldsymbol{x}_{i+1}^-}{\partial \boldsymbol{u}_i^k})^{\mathrm{T}}\frac{\partial \boldsymbol{\varPhi}(\boldsymbol{x}_{i+1}^-)}{\partial \boldsymbol{x}_{i+1}^-}\boldsymbol{W}^k = 0, \quad (18)$$

and

$$\begin{aligned}\left(\boldsymbol{W}^{k+1}\right)^{\mathrm{T}}\boldsymbol{\varPhi}(\boldsymbol{x}_i^-) =& \frac{1}{2}(\boldsymbol{u}_i^k)^{\mathrm{T}}\boldsymbol{R}\boldsymbol{u}_i^k + \frac{1}{2}(\boldsymbol{x}_i^+)^{\mathrm{T}}\bar{\boldsymbol{Q}}\boldsymbol{x}_i^+ \\ &+ (\boldsymbol{W}^k)^{\mathrm{T}}\boldsymbol{\varPhi}(\boldsymbol{x}_{i+1}^-).\end{aligned} \quad (19)$$

Eq. (18) is the first-order necessary condition for Eq. (13). In general, it is an algebraic equation of $\boldsymbol{u}_i^k$ (impulse control $\boldsymbol{u}_i$ at iteration k), which can be solved either directly or iteratively.

The least mean square (LMS) method can be used to solve Eq. (19) and find the updated network weight $\boldsymbol{W}^{k+1}$ which minimizes the residual error between the current objective function and the targeted objective function. Define the residual error as

$$\begin{aligned}e \triangleq \Big[&\frac{1}{2}(\boldsymbol{u}_i^k)^{\mathrm{T}}\boldsymbol{R}\boldsymbol{u}_i^k + \frac{1}{2}(\boldsymbol{x}_i^+)^{\mathrm{T}}\bar{\boldsymbol{Q}}\boldsymbol{x}_i^+ \\ &+(\boldsymbol{W}^k)^{\mathrm{T}}\boldsymbol{\varPhi}(\boldsymbol{x}_{i+1}^-) - (\boldsymbol{W}^{k+1})^{\mathrm{T}}\boldsymbol{\varPhi}(\boldsymbol{x}_i^-)\Big]^2.\end{aligned} \quad (20)$$

By applying the gradient based method, the network weight updating rule becomes

$$\boldsymbol{W}_{j+1}^{k+1} = \boldsymbol{W}_j^{k+1} - \alpha\frac{\partial e}{\partial \boldsymbol{W}_j^{k+1}}, \quad (21)$$

where $\alpha > 0$ is the step size, and j indexes the $\boldsymbol{W}$ iteration in the gradient based method. The step size α needs careful adjustments, so that the residual error is minimized within the desired error tolerance. Remark 4.1 in Liu and Wei (2013) gives a good explanation of the parameter selection. Other sophisticated optimization methods, such as the Newton method and conjugated gradient method, can also be applied to minimize e with respect to $\boldsymbol{W}$.

4.2 Algorithm flowchart

Fig. 2 shows the flowchart of the proposed method. The following steps are included in the algorithm:

1. Initialization: All the necessary parameters are initialized providing the known system matrices. The V function is approximated by a neural network with zero initial weights, as denoted in Eq. (17). Err is initially selected as a large number, used later to measure the change of network weights during the

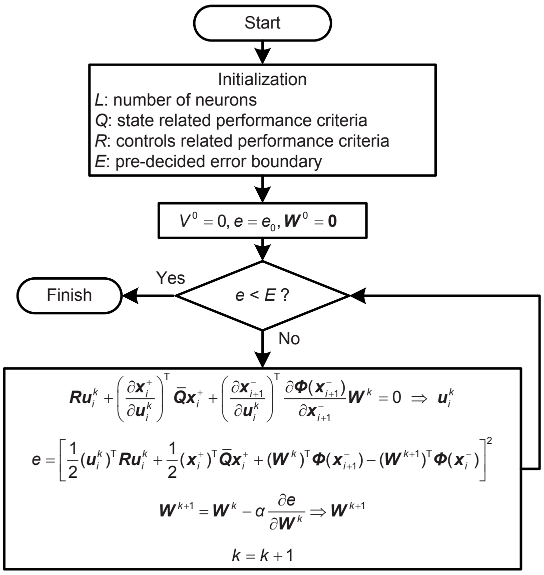

Start
Initialization
L: number of neurons
Q: state related performance criteria
R: controls related performance criteria
E: pre-decided error boundary
$V^0=0, e=e_0, \boldsymbol{W}^0=\boldsymbol{0}$
$e<E$?
Yes
Finish
No
$\boldsymbol{R}\boldsymbol{u}_i^k+\left(\frac{\partial \boldsymbol{x}_i^+}{\partial \boldsymbol{u}_i^k}\right)^{\mathrm{T}}\bar{\boldsymbol{Q}}\boldsymbol{x}_i^+ +\left(\frac{\partial \boldsymbol{x}_{i+1}^-}{\partial \boldsymbol{u}_i^k}\right)^{\mathrm{T}}\frac{\partial \boldsymbol{\Phi}(\boldsymbol{x}_{i+1}^-)}{\partial \boldsymbol{x}_{i+1}^-}\boldsymbol{W}^k=0 \Rightarrow \boldsymbol{u}_i^k$
$e=\left[\frac{1}{2}(\boldsymbol{u}_i^k)^{\mathrm{T}}\boldsymbol{R}\boldsymbol{u}_i^k+\frac{1}{2}(\boldsymbol{x}_i^+)^{\mathrm{T}}\bar{\boldsymbol{Q}}\boldsymbol{x}_i^+ +(\boldsymbol{W}^k)^{\mathrm{T}}\boldsymbol{\Phi}(\boldsymbol{x}_{i+1}^-)-(\boldsymbol{W}^{k+1})^{\mathrm{T}}\boldsymbol{\Phi}(\boldsymbol{x}_i^-)\right]^2$
$\boldsymbol{W}^{k+1}=\boldsymbol{W}^k-\alpha\frac{\partial e}{\partial \boldsymbol{W}^k}\Rightarrow \boldsymbol{W}^{k+1}$
$k=k+1$

Fig. 2 Successive approximation algorithm

iterations. The iteration index k is set as 0. The approximation domain Ω is selected to evenly cover the state space of interest.

2. Control update: The impulse control is derived minimizing the value function at the ith impulse moment, which is Eq. (13). Explicitly, Eq. (18) is used to calculate $\boldsymbol{u}_i^k$.

3. Weight update: After $\boldsymbol{u}_i^k$ is updated, function V^k can be updated according to Eq. (14). If the LMS method is used, by defining the residual error e in Eq. (20), the neural network weights are updated using Eq. (21). The error norm E is the norm of the differences between the updated weights and the previous weights.

4. Compare E with the pre-set accuracy Err. If E is greater than Err, increase the iteration index k by 1 and repeat steps 2 and 3. If E is smaller than Err, which means the weights have reached their steady states, the training is complete.

5. When the training is complete, the optimal control is derived and the optimal value function obtained, and the system dynamics with optimal control can be found accordingly.

Through the ADP method, the convergence is guaranteed as long as every update in steps 2 and 3 is successful. After the ADP training is complete, the optimal weights $\boldsymbol{W}$, optimal objective function V_i^-, and optimal impulse control $\boldsymbol{u}_i$ are obtained. The optimal control problem of the impulse systems is thus solved without the need for matrix inversion.

5 Simulations

A scalar impulse control problem and a vector problem are studied and simulated using MATLAB. The results are presented in this section.

Example 1 Consider a scalar impulse system described by

$$\dot{x}=ax+bu\delta(t-\tau_i). \tag{22}$$

The objective function follows the definition in Eq. (2). $r=1$ and $q=1$ are the weighting matrices for this scalar case. $\tau_i=1,2,3,\ldots$ are the impulse moments, where i tends to infinity.

Two sets of parameters are chosen to show the effectiveness of the network based ADP algorithm. For the first case, the parameters are chosen as $a=1, b=1, r=1$, and $q=1$. The system is unstable without an impulse control. For the second case, the system is stable without an impulse control, where $a=-1$, while the other parameters are the same. According to Lemma 1, the optimal P^- value can be decided using Eq. (11). As this is a scalar system, a direct calculation is not that difficult. For the first case, $P^-=0.9083$, and for the second case, $P^-=0.3225$. The pre-impulse optimal objective function is $\frac{1}{2}x_i^- P^- x_i^-$.

The polynomial activation vector is chosen as $\boldsymbol{\Phi}\left(x_i^-\right)=[x_i^-, \left(x_i^-\right)^2]^{\mathrm{T}}$. The corresponding weight vector is $\boldsymbol{W}=[W_1, W_2]^{\mathrm{T}}$. For ADP network training, the pre-impulse state space is randomly chosen with five elements. Note that the number of states included for training the neural network must be greater than the number of weights to be tuned. For $\boldsymbol{W}\in\mathbb{R}^{2\times 1}$, at least two elements should be selected in the pre-impulse state space for tuning the weights.

When $a=1, b=1, r=1$, and $q=1$, Fig. 3 shows the change of network weights $\boldsymbol{W}$ during the iterations. Fig. 4 depicts the evolution of the objective function V_i^- for each of the x_i^- in the training set. Both $\boldsymbol{W}$ and V_i^- go to their steady states in the figures. $\boldsymbol{W}$ tends to $[0, 0.4542]^{\mathrm{T}}$, which essentially represents the same objective function as $P^-=0.9083$.

When $a=-1, b=1, r=1$, and $q=1$, Fig. 5 shows the change of network weights $\boldsymbol{W}$ during the iterations. Fig. 6 depicts the evolvement of the objective function V_i^- for each of the x_i^- in the training set. Both $\boldsymbol{W}$ and V_i^- go to their steady states in the figures. $\boldsymbol{W}$ tends to $[0, 0.1612]^{\mathrm{T}}$, which is essentially the same as $P^{-} = 0.3225$.

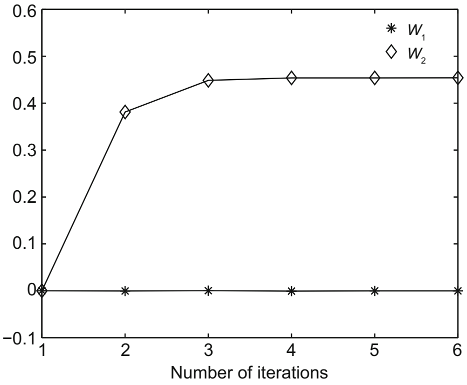


Fig. 3 Iterative process of the network weights for $a = 1, b = 1, r = 1$, and $q = 1$

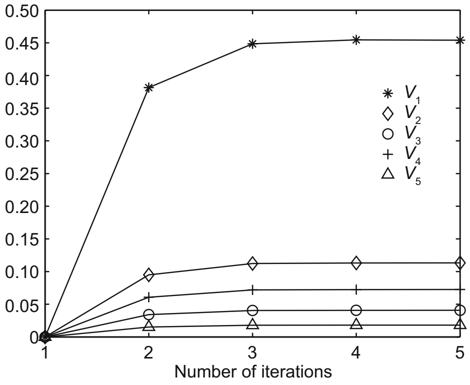


Fig. 4 Iterative process of the pre-impulse objective functions for $a = 1, b = 1, r = 1$, and $q = 1$

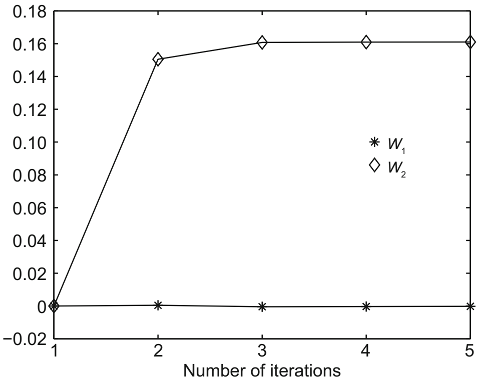


Fig. 5 Iterative process of the network weights for $a = -1, b = 1, r = 1$, and $q = 1$

From Figs. 3–6, it can be seen that the neural network weights tend to be the optimal ones in the steady state after about five iterations. The optimal objective function is captured by the trained network using the proposed ADP method.

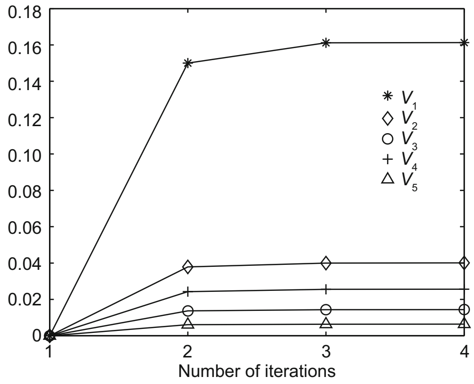


Fig. 6 Iterative process of the pre-impulse objective functions for $a = -1, b = 1, r = 1$, and $q = 1$

Example 2 Consider a 2D vector system described by

$$\dot{\boldsymbol{x}} = \begin{bmatrix} 0 & 1 \\ 1 & -5 \end{bmatrix} \boldsymbol{x} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} \boldsymbol{u} \delta \left(t - \tau_i \right). \tag{23}$$

The quadratic objective function to be minimized has a state weighting matrix $\boldsymbol{Q} = \boldsymbol{I}_{2\times 2}$ and a control weighting matrix $\boldsymbol{R} = \boldsymbol{I}_{1\times 1}$.

Assuming fixed impulsive moments at $\tau_i = 1, 2, 3, \cdots$, the optimal $\boldsymbol{P}^{-}$ matrix calculated using Eq. (11) is

$$\boldsymbol{P}^{-} = \begin{bmatrix} 10.8813 & 1.8937 \\ 1.8937 & 0.4204 \end{bmatrix}. \tag{24}$$

Select the activation function vector as $\boldsymbol{\Phi}(\boldsymbol{X}) = [x_1, x_2, x_1^2, 2x_1x_2, x_2^2]$, where x_1 and x_2 are the first and second system states, respectively. The pre-impulse objective function is then expressed as $\boldsymbol{W}^{\mathrm{T}}\boldsymbol{\Phi}(\boldsymbol{X}_i^{-})$, where $\boldsymbol{W} = [W_1, W_2, W_3, W_4, W_5]^{\mathrm{T}}$.

Follow the ADP method of iterative training using Eqs. (18) and (19). For this example, since $\boldsymbol{W} \in \mathbb{R}^5$, five weights need to be optimized; therefore, the number of pre-impulse states should be no less than five. Five pre-impulse states are randomly chosen. Fig. 7a depicts the evolvement of the objective functions for the five pre-impulse states. Fig. 7b depicts the iterative process of the weights with different marks.

Figs. 7a and 7b show that the weights and objective functions converge to the steady states after about 13 iterations. In the simulation, the steady state network weights $\boldsymbol{W}$=[0.0004, 0.0004, 5.4399, 0.9468, 0.2103]$^{\mathrm{T}}$. The weights essentially represent the same objective as matrix $\boldsymbol{P}$ does in Eq. (24).

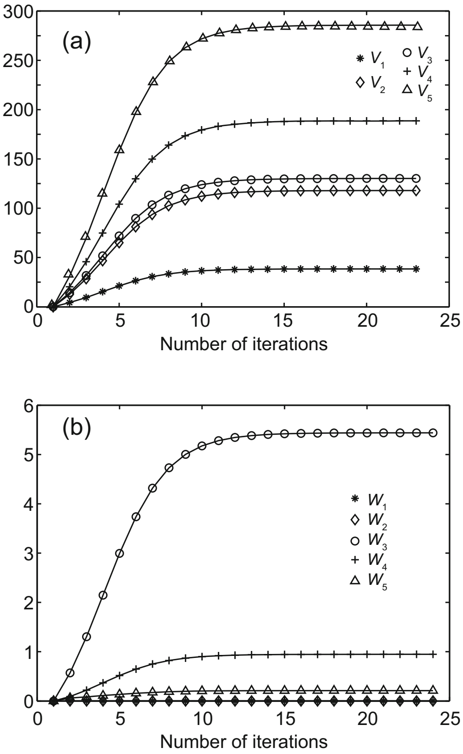


Fig. 7 Iterative process of the pre-impulse objective functions (a) and the neural network weights (b) for the vector case

In both scalar and vector cases, the ADP algorithm using neural network approximation provides results converging to optimal control and avoids matrix inversion.

6 Conclusions

In this paper, a neural network based adaptive dynamic programming (ADP) method is proposed to derive the optimal impulse control. This method provides optimal impulse control in a feedback form. The neural network is used to iteratively approximate the objective function in the ADP algorithm. The convergence of the ADP algorithm has been shown. Through neural network approximation, the matrix inversion is avoided in the ADP using the direct quadratic value function of the states. For high order systems, the numerical singularity problem caused by matrix inversions can be avoided. Simulation examples have illustrated the validity of the proposed method. In the future, it would be interesting to show how to derive the optimal impulse control when the system dynamics $\boldsymbol{A}$ and $\boldsymbol{B}$ are unknown.

References

Ahmed, N.U., 2003. Existence of optimal controls for a general class of impulsive systems on Banach spaces. *SIAM J. Control Optim.*, **42**(2):669-685. [doi:10.1137/S0363012901391299]

Bainov, D.D., Simeonov, P.S., 1995. Impulsive Differential Equations: Asymptotic Properties of the Solutions. World Scientific, Singapore.

Balakrishnan, S.N., Ding, J., Lewis, F.L., 2008. Issues on stability of ADP feedback controllers for dynamical systems. *IEEE Tran. Syst. Man Cybern. B*, **38**(4):913-917. [doi:10.1109/TSMCB.2008.926599]

Bertsekas, D.P., 2011. Approximate policy iteration: a survey and some new methods. *J. Control Theory Appl.*, **9**(3):310-335.

Dierks, T., Jagannathan, S., 2011. Online optimal control of nonlinear discrete-time systems using approximate dynamic programming. *J. Control Theory Appl.*, **9**(3):361-369. [doi:10.1007/s11768-011-0178-0]

Fraga, S.L., Pereira, F.L., 2012. Hamilton-Jacobi-Bellman equation and feedback synthesis for impulsive control. *IEEE Trans. Automat. Control*, **57**(1):244-249. [doi:10.1109/TAC.2011.2167822]

Jiang, Y., Jiang, Z.P., 2012. Computational adaptive optimal control for continuous-time linear systems with completely unknown dynamics. *Automatica*, **48**(10):2699-2704. [doi:10.1016/j.automatica.2012.06.096]

Jiang, Z.P., Jiang, Y., 2013. Robust adaptive dynamic programming for linear and nonlinear systems: an overview. *Eur. J. Control*, **19**(5):417-425. [doi:10.1016/j.ejcon.2013.05.017]

Kurzhanski, A.B., Daryin, A.N., 2008. Dynamic programming for impulse controls. *Ann. Rev. Control*, **32**(2):213-227. [doi:10.1016/j.arcontrol.2008.08.001]

Lakshmikantham, V., Bainov, D.D., Simeonov, P.S., 1989. Theory of Impulsive Differential Equations. World Scientific, Singapore.

Lewis, F.L., Vrabie, D., 2009. Reinforcement learning and adaptive dynamic programming for feedback control. *IEEE Circ. Syst. Mag.*, **9**(3):32-50. [doi:10.1109/MCAS.2009.933854]

Liu, B., Teo, K.L., Liu, X.Z., 2008. Optimal control and robust stability of uncertain impulsive dynamical systems. *Asian J. Control*, **10**(3):314-326. [doi:10.1002/asjc.37]

Liu, D.R., Wei, Q.L., 2013. Finite-approximation-error-based optimal control approach for discrete-time nonlinear systems. *IEEE Trans. Cybern.*, **43**(2):779-789. [doi:10.1109/TSMCB.2012.2216523]

Liu, X., 1995. Impulsive control and optimization. *Appl. Math. Comput.*, **73**(1):77-98. [doi:10.1016/0096-3003(94)00204-H]

Silva, G.N., Vinter, R.B., 1997. Necessary conditions for optimal impulsive control problems. *SIAM J. Control Optim.*, **35**(6):1829-1846. [doi:10.1137/S0363012995281857]

Vamvoudakis, K.G., Lewis, F.L., 2010. Online actor–critic algorithm to solve the continuous-time infinite horizon optimal control problem. *Automatica*, **46**(5):878-888. [doi:10.1016/j.automatica.2010.02.018]

Wang, F.Y., Zhang, H.G., Liu, D.R., 2009. Adaptive dynamic programming: an introduction. *IEEE Comput. Intell. Mag.*, **4**(2):39-47. [doi:10.1109/MCI.2009.932261]

Wang, J.R., Yang, Y.L., 2010. Optimal control of linear impulsive antiperiodic boundary value problem on infinite dimensional spaces. *Discr. Dynam. Nat. Soc.*, Article ID 673013. [doi:10.1155/2010/673013]

Wang, X.H., 2008. Optimal Control of Impulsive Systems Using Adaptive Critic Neural Network. PhD Thesis, Missouri University of Science and Technology, Rolla, Missouri, USA.

Wang, X.H., Balakrishnan, S.N., 2010. Optimal neurocontroller synthesis for impulse-driven systems. *Neur. Networks*, **23**(1):125-134. [doi:10.1016/j.neunet.2009.08.009]

Wang, X.H., Luo, W.Z., Balakrishnan, S.N., 2012. Linear impulsive system optimization using adaptive dynamic programming. 12th Int. Conf. on Control Automation Robotics and Vision, p.725-730. [doi:10.1109/ICARCV.2012.6485247]

Werbos, P.J., 1974. Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences. PhD Thesis, Harvard University, USA.

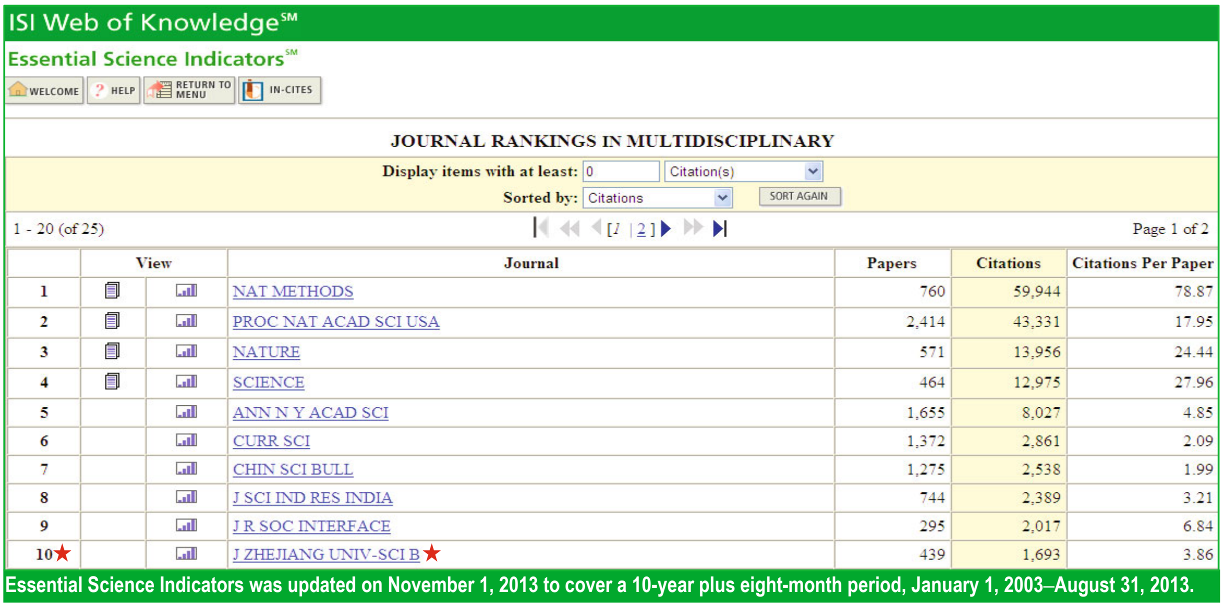
Werbos, P.J., 2008. Foreword-ADP: the key direction for future research in intelligent control and understanding brain intelligence. *IEEE Trans. Syst. Man Cybern. B*, **38**(4):898-900. [doi:10.1109/TSMCB.2008.924139]

Werbos, P.J., McAvoy, T., Su, T., 1992. Handbook of Intelligent Control. Van Nostrand Reinhold, New York.

Wu, Z., Zhang, F., 2011. Stochastic maximum principle for optimal control problems of forward-backward systems involving impulse controls. *IEEE Trans. Automat. Control*, **56**(6):1401-1406.

Yang, T., 1999. Impulsive control. *IEEE Trans. Automat. Control*, **44**(5):1081-1083.

ESI Journal Ranking: *J ZHEJIANG UNIV-SCI B* Ranks No. 10 in Multidisciplinary



ISI Web of Knowledge℠

Essential Science Indicators℠

JOURNAL RANKINGS IN MULTIDISCIPLINARY

Display items with at least: 0 Citation(s)

Sorted by: Citations

1 - 20 (of 25)

Page 1 of 2

	View		Journal	Papers	Citations	Citations Per Paper
1	📄	📊	NAT METHODS	760	59,944	78.87
2	📄	📊	PROC NAT ACAD SCI USA	2,414	43,331	17.95
3	📄	📊	NATURE	571	13,956	24.44
4	📄	📊	SCIENCE	464	12,975	27.96
5		📊	ANN N Y ACAD SCI	1,655	8,027	4.85
6		📊	CURR SCI	1,372	2,861	2.09
7		📊	CHIN SCI BULL	1,275	2,538	1.99
8		📊	J SCI IND RES INDIA	744	2,389	3.21
9		📊	J R SOC INTERFACE	295	2,017	6.84
10★		📊	J ZHEJIANG UNIV-SCI B★	439	1,693	3.86

Essential Science Indicators was updated on November 1, 2013 to cover a 10-year plus eight-month period, January 1, 2003–August 31, 2013.