Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2014, Article ID 628798, 16 pages
http://dx.doi.org/10.1155/2014/628798

Research Article

Discrete Globalised Dual Heuristic Dynamic Programming in Control of the Two-Wheeled Mobile Robot

Marcin Szuster and Zenon Hendzel

Department of Applied Mechanics and Robotics, Rzeszow University of Technology, 8 Powstancow Warszawy Street, 35-959 Rzeszow, Poland

Correspondence should be addressed to Marcin Szuster; mszuster@prz.edu.pl

Received 8 April 2014; Revised 17 August 2014; Accepted 17 August 2014; Published 14 September 2014

Academic Editor: Yang Xu

Network-based control systems have been emerging technologies in the control of nonlinear systems over the past few years. This paper focuses on the implementation of the approximate dynamic programming algorithm in the network-based tracking control system of the two-wheeled mobile robot, Pioneer 2-DX. The proposed discrete tracking control system consists of the globalised dual heuristic dynamic programming algorithm, the PD controller, the supervisory term, and an additional control signal. The structure of the supervisory term derives from the stability analysis realised using the Lyapunov stability theorem. The globalised dual heuristic dynamic programming algorithm consists of two structures: the actor and the critic, realised in a form of neural networks. The actor generates the suboptimal control law, while the critic evaluates the realised control strategy by approximation of value function from the Bellman's equation. The presented discrete tracking control system works online, the neural networks' weights adaptation process is realised in every iteration step, and the neural networks preliminary learning procedure is not required. The performance of the proposed control system was verified by a series of computer simulations and experiments realised using the wheeled mobile robot Pioneer 2-DX.

1. Introduction

A rapid development of the mobile robotics applications in the last few years can be observed. Autonomous wheeled mobile robots (WMRs) have attracted much attention among researchers and engineers, while construction of robots, their sensory systems, and control algorithms were developed. One of the most challenging tasks, which occurs in the implementations of autonomous WMR, is the tracking control problem. It is widely discussed in literature, where different control strategies [1–4] are presented. This shows how significant the problem is. Difficulties met in the realisation of the desired trajectory by WMRs result from the fact that these control objects are described using nonlinear dynamic equations, where some parameters of the model can be unknown or change during the movement, for the sake of disturbances. This results in the necessity of application of computationally complex methods, which can adjust their parameters during the realisation of the trajectory and assure required quality of tracking. Artificial intelligence (AI) methods, like neural networks (NNs) [1, 2, 5, 6], are willingly applied in control systems of robots, for the sake of weights adaptation possibility. The development of AI methods makes the implementation of Bellman's dynamic programming (DP) [7] idea possible. This group of methods is called approximated dynamic programming algorithms (ADP) [8–12], adaptive critic designs (ACD), neurodynamic programming algorithms, or actor-critic structures. It is included in the larger family of methods adapted using the reinforcement learning (RL) idea. According to [9, 12], the ADP algorithms family is composed of six main schemes: heuristic dynamic programming (HDP), dual heuristic dynamic programming (DHP), globalised DHP (GDHP), and action dependant versions of mentioned earlier algorithms: action-dependent HDP (ADHDP), ADDHP, and ADGDHP. Very good surveys on ADP are given in [9, 13–16]. ADP algorithms have been firstly described for discrete-time systems [8, 9, 12] and few years later, for time-continuous systems [17–21].

Simultaneously with continuous high interest in RL algorithms, a growing number of its applications can be observed. The challenging applications of RL methods are the control problems of autonomous robots like the helicopter [22] or the underwater vehicle [23]. There are implementations of RL algorithms in mobile robot path planning [24], urban traffic signal control [25], or power system control [26], but these are mostly implementations of the Q-learning algorithm [10]. There are not many recent articles concerning ADP algorithms; the example is the application of ADHDP algorithm for a static compensator connected to a power system [27] or HDP and DHP algorithms in target recognition [28]. Application of the ADP algorithms in the control of the wheeled mobile robot is presented in [4] and in the trajectory generating process in [29]. In [30, 31] the HDP algorithm is applied to the control of the nonlinear system with some simulation results. Interesting results are shown in [32], where based on the HDP and the DHP algorithms, new kernel versions were proposed that can obtain better performance than original ones. The performance was tested using the inverted pendulum and the ball and plate benchmark systems. The implementation of the GDHP algorithm for the control of the linear object is described in [33] and for the control of the nonlinear system in [3, 34, 35], the control problem of the turbo-generator, solved using this algorithm, is presented in [36]. The article [37] summarizes the novel developments in policy-gradient and presents the novel RL architecture, the natural actor-critic (NAC), and the simulation test performed in the cart-pole balancing problem. Recent works on ADP algorithms have attempted to solve the problem of implementation of ADP based control systems without a system model knowledge [17–19]. Recent advances in this field also include implementation of ADP algorithms for partially unknown nonlinear systems [19] and robust optimal tracing control for the unknown nonlinear system [38].

The paper presents the application of the ADP algorithm in the GDHP configuration [3, 33–35] in the tracking control problem of the WMR. The discrete tracking control system guarantees a high tracking performance and a stable realisation of the desired trajectory in the face of disturbances. The GDHP algorithm consists of two structures, the actor and the critic, both realised in the form of random vector functional link (RVFL) NNs [2]. Solutions of the tracking control problems presented in literature are often theoretical considerations; there are not many real applications of ADP algorithms in control problems. The proposed discrete neural tracking control system is used for the tracking control of the WMR Pioneer 2-DX, where a series of computer simulations and experiments were realised to illustrate the performance of the control algorithm.

The results of the research presented in the paper continue the authors' earlier works related to the problem of control of the ball and beam systems [39] and the robotic manipulator [40] using DHP algorithm, tracking control of the WMR [41–44] using different ADP algorithms, and the problem of trajectory generating using ADHDP [45]. The remainder of this paper is organised as follows. The WMR dynamics is given in Section 2. The ADP algorithms family is described in Section 3. In Section 4 the GDHP algorithm implemented in the proposed discrete tracking control system is presented and in the following section, the stability is analysed using the Lyapunov function. In Section 6, the effectiveness of the proposed control algorithm is demonstrated through a numerical illustration and an experiment realised using the WMR Pioneer 2-DX. Finally, Section 7 gives the conclusion.

2. Dynamical Model of the Wheeled Mobile Robot Pioneer 2-DX

The WMR Pioneer 2-DX is the control object, shown in Figure 1(a). It is a nonholonomic object, which dynamics is described using nonlinear equations. The WMR is composed of two driving wheels 1 and 2, a third, free rolling castor wheel 3, and a frame 4 (Figure 1(b)). The movement of the WMR is analysed in the xy plane.

Point A is a central point of the WMR's frame, β is an angle of the frame's turn, r_1, r_2, l, and l_1 are dimensions that result from the WMR's geometry, $\alpha_{[1]}$, $\alpha_{[2]}$ are angles of the driving wheels 1 and 2 rotation, and $u_{[1]}$, $u_{[2]}$ are control signals. The dynamical model of the WMR was derived using Maggie's formalism [2, 46] and assumed in the form

$$\mathbf{M}\ddot{\boldsymbol{\alpha}} + \mathbf{C}(\dot{\boldsymbol{\alpha}})\dot{\boldsymbol{\alpha}} + \mathbf{F}(\dot{\boldsymbol{\alpha}}) + \boldsymbol{\tau}_d(t) = \mathbf{u}, \tag{1}$$

where $\dot{\boldsymbol{\alpha}} = [\dot{\alpha}_{[1]}, \dot{\alpha}_{[2]}]^T$ is the vector of angular velocities of driving wheels, $\mathbf{M}$ is the positive defined inertia matrix, $\mathbf{C}(\dot{\boldsymbol{\alpha}})\dot{\boldsymbol{\alpha}}$ is the vector of centrifugal and Coriolis forces/momentous, $\mathbf{F}(\dot{\boldsymbol{\alpha}})$ is the friction vector, $\boldsymbol{\tau}_d(t)$ is the vector of disturbances, and $\mathbf{u}$ is the control vector. Matrices $\mathbf{M}$, $\mathbf{C}(\dot{\boldsymbol{\alpha}})$ and the vector $\mathbf{F}(\dot{\boldsymbol{\alpha}})$ take the form

$$\mathbf{M} = \begin{bmatrix} a_{[1]} + a_{[2]} + a_{[3]} & a_{[1]} - a_{[2]} \\ a_{[1]} - a_{[2]} & a_{[1]} + a_{[2]} + a_{[3]} \end{bmatrix},$$

$$\mathbf{C}(\dot{\boldsymbol{\alpha}}) = \begin{bmatrix} 0 & 2a_{[2]}\left(\dot{\alpha}_{[2]} - \dot{\alpha}_{[1]}\right) \\ -2a_{[2]}\left(\dot{\alpha}_{[2]} - \dot{\alpha}_{[1]}\right) & 0 \end{bmatrix}, \tag{2}$$

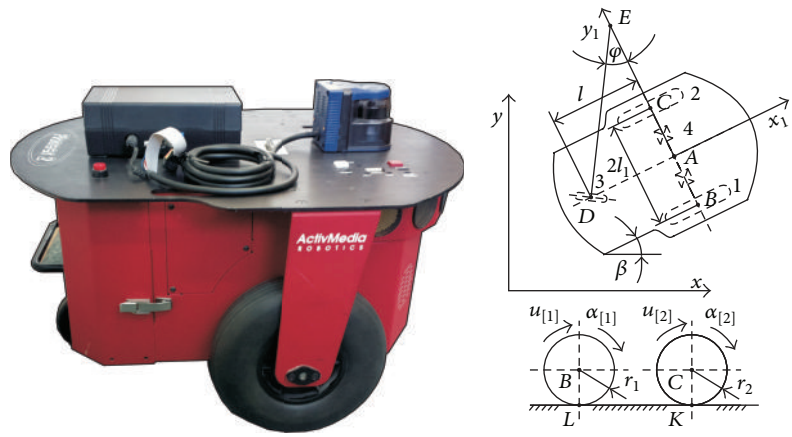
$$\mathbf{F}(\dot{\boldsymbol{\alpha}}) = \begin{bmatrix} a_{[5]}\operatorname{sgn}\left(\dot{\alpha}_{[1]}\right) \\ a_{[6]}\operatorname{sgn}\left(\dot{\alpha}_{[2]}\right) \end{bmatrix},$$

where $\mathbf{a} = [a_{[1]}, \ldots, a_{[6]}]^T$ is the vector of WMR's parameters that result from the object's geometry, mass distribution, and resistances to motion [2, 46]. The nominal parameters of the WMR Pioneer 2-DX were assumed as $\mathbf{a} = [0.1207, 0.0768, 0.037, 0.0001, 2.025, 2.025]^T$.

The proposed tracking control system is discrete. A continuous model of the WMR's dynamics (1) was discretised using Euler's method and assumed in the form

$$\mathbf{z}_{1\{k+1\}} = \mathbf{z}_{1\{k\}} + h\mathbf{z}_{2\{k\}},$$

$$\mathbf{z}_{2\{k+1\}} = \mathbf{z}_{2\{k\}} - h\mathbf{M}^{-1}\left[\mathbf{C}\left(\mathbf{z}_{2\{k\}}\right)\mathbf{z}_{2\{k\}} + \mathbf{F}\left(\mathbf{z}_{2\{k\}}\right) + \boldsymbol{\tau}_{d\{k\}} - \mathbf{u}_{\{k\}}\right], \tag{3}$$



(a) (b)

FIGURE 1: (a) The WMR Pioneer 2-DX, (b) scheme of the WMR.

where $\mathbf{z}_{2\{k\}} = [z_{2[1]\{k\}}, z_{2[2]\{k\}}]^T$ is a vector that corresponds to the continuous vector of angular velocities $\dot{\boldsymbol{\alpha}}$, k is an index of iteration steps, and h is a time discretisation parameter. The state vector was assumed in the form $\mathbf{z}_{\{k\}} = [\mathbf{z}_{1\{k\}}, \mathbf{z}_{2\{k\}}]^T$. The discrete tracking errors of angles of the driving wheels rotation $\mathbf{z}_{1\{k\}}$ and errors of angular velocities $\mathbf{z}_{2\{k\}}$ were defined as

$$\begin{aligned} \mathbf{e}_{1\{k\}} &= \mathbf{z}_{1\{k\}} - \mathbf{z}_{d1\{k\}}, \\ \mathbf{e}_{2\{k\}} &= \mathbf{z}_{2\{k\}} - \mathbf{z}_{d2\{k\}}, \end{aligned} \tag{4}$$

where the desired trajectory ($\mathbf{z}_{d\{k\}} = [\mathbf{z}_{d1\{k\}}, \mathbf{z}_{d2\{k\}}]^T$) was generated earlier. On the basis of (4) the filtered tracking error $\mathbf{s}_{\{k\}}$ was defined in the form

$$\mathbf{s}_{\{k\}} = \mathbf{e}_{2\{k\}} + \boldsymbol{\Lambda}\mathbf{e}_{1\{k\}}, \tag{5}$$

where $\boldsymbol{\Lambda}$ is a positive defined, fixed diagonal matrix.

Substituting the WMR dynamics model (3) and the tracking errors (4) into $\mathbf{s}_{\{k+1\}}$, calculated on the base of (5), the filtered tracking error was assumed in the form

$$\begin{aligned} &\mathbf{s}_{\{k+1\}} \\ &= \mathbf{Y}_d\left(\mathbf{z}_{\{k\}}, \mathbf{z}_{d\{k\}}, \mathbf{z}_{d3\{k\}}\right) - \mathbf{Y}_f\left(\mathbf{z}_{2\{k\}}\right) - \mathbf{Y}_{\tau\{k\}} + h\mathbf{M}^{-1}\mathbf{u}_{\{k\}}, \end{aligned} \tag{6}$$

where

$$\begin{aligned} &\mathbf{Y}_d\left(\mathbf{z}_{\{k\}}, \mathbf{z}_{d\{k\}}, \mathbf{z}_{d3\{k\}}\right) \\ &\quad = \mathbf{z}_{2\{k\}} - \mathbf{z}_{d2\{k+1\}} + \boldsymbol{\Lambda}\left[\mathbf{z}_{1\{k\}} + h\mathbf{z}_{2\{k\}} - \mathbf{z}_{d1\{k+1\}}\right] \\ &\quad = \mathbf{s}_{\{k\}} + \mathbf{Y}_e\left(\mathbf{z}_{2\{k\}}, \mathbf{z}_{d2\{k\}}, \mathbf{z}_{d3\{k\}}\right), \\ &\mathbf{Y}_e\left(\mathbf{z}_{2\{k\}}, \mathbf{z}_{d2\{k\}}, \mathbf{z}_{d3\{k\}}\right) = h\left[\boldsymbol{\Lambda}\mathbf{e}_{2\{k\}} - \mathbf{z}_{d3\{k\}}\right], \\ &\mathbf{Y}_f\left(\mathbf{z}_{2\{k\}}\right) = h\mathbf{M}^{-1}\left[\mathbf{C}\left(\mathbf{z}_{2\{k\}}\right)\mathbf{z}_{2\{k\}} + \mathbf{F}\left(\mathbf{z}_{2\{k\}}\right)\right], \\ &\mathbf{Y}_{\tau\{k\}} = h\mathbf{M}^{-1}\boldsymbol{\tau}_{d\{k\}}, \end{aligned} \tag{7}$$

where $\mathbf{z}_{d3\{k\}}$ is the vector of desired angular accelerations that derives from the expansion of the vector $\mathbf{z}_{d2\{k+1\}}$ using Euler's method. The vector $\mathbf{Y}_f(\mathbf{z}_{2\{k\}})$ includes all nonlinearities of the controlled object.

3. Approximate Dynamic Programming

Bellman's dynamic programming (DP) is based on the calculation of the value function, the control law, and the state of the object for every step of the process, from the last to the first. That is why it is not applicable in online control. ADP algorithms are also called adaptive critic designs (ACD) [8–16] or neuro-dynamic programming (NDP) algorithms. They derive from the application of NNs into Bellman's approach to the optimal control theory, where the value function and the optimal control law are approximated by the critic and the actor. This approach makes real-time control of dynamical objects possible. The ADP algorithms family is schematically shown in Figure 2. It is composed of six algorithms, which differ from each other by the critic's structure and the weights adaptation rule of the actor's and the critic's NN.

The basic structure is the HDP algorithm, in which the critic approximates the value function and the actor generates the suboptimal control law. In the DHP algorithm the critic approximates the difference of the value function with respect to the state of the controlled system. The actor has the same structure as in HDP. Complexity of the critic grows proportionally to the size of the state vector, because the difference of the value function with respect to the n-dimensional state vector is approximated by n critic's NNs, and the critic's weights adaptation law is also more complex. The DHP algorithm assures higher quality of tracking control in comparison to HDP [43]. The GDHP algorithm is built in the same way as HDP; its characteristic feature is the critic's weights adaptation law. It is based on the minimisation of the value function and its difference with respect to the state and can be seen as a combination of the HDP and the DHP critic's NN adaptation law. The actor structure is the same as

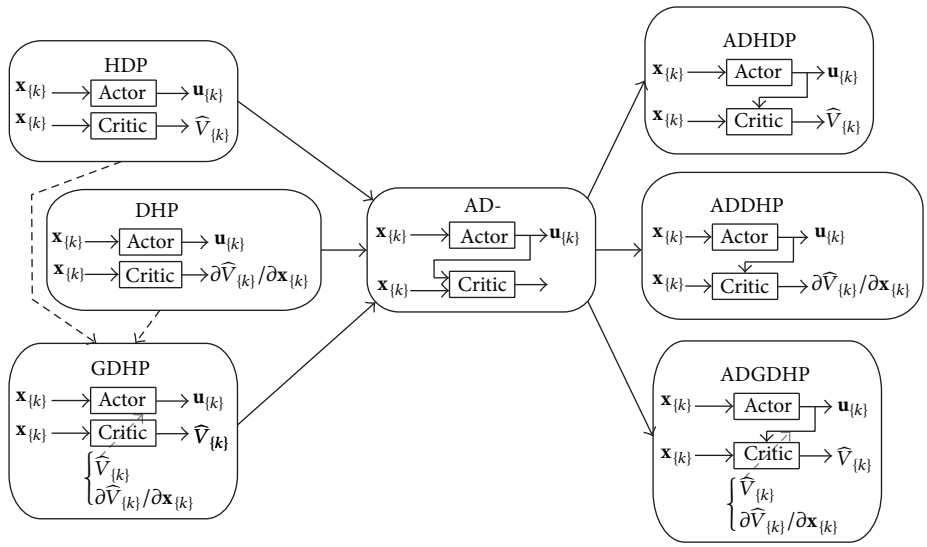


Figure 2: The scheme of the approximate dynamic programming algorithms family.

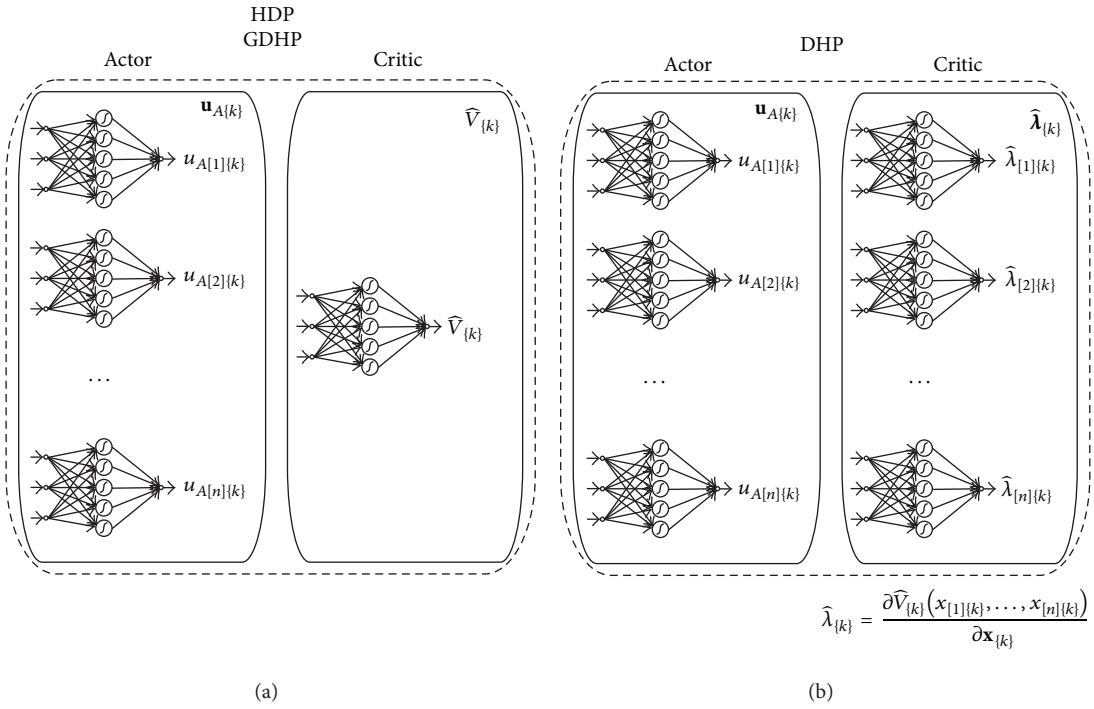


Figure 3: (a) Scheme of the actor's and the critic's structure complexity in HDP and GDHP, (b) scheme of the actor's and the critic's structure complexity in DHP.

in HDP. The difference in complexity of the three basic ADP algorithms is schematically shown in Figure 3.

In the HDP and the GDHP algorithm the critic is composed of one NN that approximates the value function, while in the DHP algorithm critic consists of n NNs, where n is the size of the state vector. For example, in the case of the WMR, where the state vector for the system (6) is of $n = 2$ size, the DHP algorithm consists of the actor and the critic realised in a form of two NNs each. In the GDHP algorithm, the actor is composed of two NNs, but the critic is realised in the form of only one NN. The advantage of GDHP over DHP, in the case of complexity of the critic, is even more evident considering the instance of the 6 degrees of freedom robotic manipulator ($n = 6$). The DHP algorithm implemented in the control system for this controlled object should be composed of the actor and the critic realised in a form of six NNs each, while the GDHP would be composed of the actor realised in a form of six NNs, and only one NN in the critic structure. The

difference of the complexity of the critic structure increases simultaneously as the state vector of the controlled object increases. The rest of the ADP algorithms are AD versions of the basic algorithms, where the control law generated by the actor's NN is also the input to the critic's NN.

4. Globalised Dual Heuristic Dynamic Programming in Tracking Control

The main part of the proposed tracking control system is the GDHP algorithm. There are not many applications of the GDHP algorithms in literature, and existing publications concern rather with theoretical studies [3, 33–36]. In this paper, both the numerical tests and the verification experiments of the neural tracking control system, realised using the WMR Pioneer 2-DX, are presented. The GDHP structure generates the control law that minimises the value function $V_{\{k\}}(\mathbf{s}_{\{k\}})$ [8–16], assumed in the form of equation

$$V_{\{k\}}\left(\mathbf{s}_{\{k\}}\right) = \sum_{k=0}^{N} \gamma^k L_{C\{k\}}\left(\mathbf{s}_{\{k\}}\right), \tag{8}$$

where N is a number of iteration steps, γ is a discount factor, $0 < \gamma \leq 1$, and $L_{C\{k\}}(\mathbf{s}_{\{k\}})$ is the local cost function for the kth step, assumed in the form

$$L_{C\{k\}}\left(\mathbf{s}_{\{k\}}\right) = \frac{1}{2}\mathbf{s}_{\{k\}}^T \mathbf{R}\mathbf{s}_{\{k\}}, \tag{9}$$

where $\mathbf{R}$ is a positive defined, fixed diagonal matrix.

The GDHP algorithm, schematically shown in Figure 4(a), consists of the following:

(i) the predictive model that predicts the WMR's closed-loop state $\mathbf{s}_{\{k+1\}}$, according to the equation

$$\mathbf{s}_{\{k+1\}} = \mathbf{Y}_d\left(\mathbf{z}_{\{k\}}, \mathbf{z}_{d\{k\}}, \mathbf{z}_{d3\{k\}}\right) - \mathbf{Y}_f\left(\mathbf{z}_{2\{k\}}\right) + h\mathbf{M}^{-1}\mathbf{u}_{\{k\}}, \tag{10}$$

where $\mathbf{u}_{\{k\}}$ is the overall tracking control signal of the proposed control system. Its structure derives from the stability analysis presented in the next section. The controlled system's dynamical model is necessary in the synthesis of the actor's and the critic's weights adaptation law in the GDHP algorithm;

(ii) the actor, realised in the form of two RVFL NNs, that generate the suboptimal control law $\mathbf{u}_{A\{k,l\}} = [u_{A[1]\{k,l\}}, u_{A[2]\{k,l\}}]^T$ and are expressed by the formula

$$u_{A[j]\{k,l\}}\left(\mathbf{x}_{Aj\{k\}}, \mathbf{W}_{Aj\{k,l\}}\right) = \mathbf{W}_{Aj\{k,l\}}^T \mathbf{S}\left(\mathbf{D}_A^T \mathbf{x}_{Aj\{k\}}\right), \tag{11}$$

where $j = 1, 2, l$ is an index of the internal loop iteration, $\mathbf{x}_{Aj\{k\}}$ is the input vector of the jth actor's NN, it consists of normalised values of the filtered tracking error $\mathbf{s}_{\{k\}}$, errors $\mathbf{e}_{\{k\}}$, desired ($\mathbf{z}_{d2\{k\}}$) and realised ($\mathbf{z}_{2\{k\}}$) angular velocities of the driving wheels, $x_{Aj[i]\{k\}} \epsilon \langle -1; 1\rangle$, $\mathbf{W}_{Aj\{k,l\}}$ is the vector of output layer weights of the jth actor's NN, $\mathbf{S}(\cdot)$ is the vector of sigmoidal bipolar neuron activation functions, and $\mathbf{D}_A$ is the matrix of fixed input weights selected randomly in the NNs initialisation process. Actor's NNs weights are adapted by the gradient method according to equation

$$\mathbf{W}_{Aj\{k,l+1\}} = \mathbf{W}_{Aj\{k,l\}} - e_{A[j]\{k,l\}} \boldsymbol{\Gamma}_A \mathbf{S}\left(\mathbf{D}_A^T \mathbf{x}_{Aj\{k\}}\right), \tag{12}$$

where $\boldsymbol{\Gamma}_A$ is the fixed diagonal matrix of positive learning rates. The quality rating $\mathbf{e}_{A\{k,l\}}$ was assumed in the form

$$\mathbf{e}_{A\{k,l\}} = \frac{\partial L_{C\{k\}}\left(\mathbf{s}_{\{k\}}\right)}{\partial \mathbf{u}_{\{k\}}} + \left[\frac{\partial \mathbf{s}_{\{k+1\}}}{\partial \mathbf{u}_{\{k\}}}\right]^T \frac{\partial \widehat{V}_{\{k+1,l\}}\left(\mathbf{x}_{C\{k+1\}}, \mathbf{W}_{C\{k,l\}}\right)}{\partial \mathbf{s}_{\{k+1\}}}, \tag{13}$$

where $\widehat{V}_{\{k+1,l\}}(\mathbf{x}_{C\{k+1\}}, \mathbf{W}_{C\{k,l\}})$ is the output of the critic's NN, generated on the basis of the predicted state for the step $k+1$;

(iii) the critic, realised in the form of one RVFL NN, estimates the value function (8). It is expressed by the formula

$$\widehat{V}_{\{k,l\}}\left(\mathbf{x}_{C\{k\}}, \mathbf{W}_{C\{k,l\}}\right) = \mathbf{W}_{C\{k,l\}}^T \mathbf{S}\left(\mathbf{D}_C^T \mathbf{x}_{C\{k\}}\right), \tag{14}$$

where $\mathbf{x}_{C\{k\}}$ is the input vector of the critic's NN, $\mathbf{x}_{C\{k\}} = \boldsymbol{\kappa}_C[1, \mathbf{s}_{\{k\}}^T]^T$, $\boldsymbol{\kappa}_C$ is the constant diagonal matrix of positive scaling coefficients, $\mathbf{W}_{C\{k,l\}}$ is the vector of output layer weights of the critic's NN, and $\mathbf{D}_C$ is the matrix of fixed input weights selected randomly in the critic's NN initialisation process. The critic's RVFL NN is schematically shown in Figure 4(b).

The critic's weights adaptation procedure in the GDHP algorithm is the most complex among all the ADP structures family. It is based on the minimisation of errors characteristic for the critic's weights adaptation rule of the HDP algorithm ($e_{H\{k,l\}}$) and the DHP algorithm ($e_{D\{k,l\}}$), expressed by the formula

$$\begin{aligned} e_{H\{k,l\}} = {} & \widehat{V}_{\{k,l\}}\left(\mathbf{x}_{C\{k\}}, \mathbf{W}_{C\{k,l\}}\right) - L_{C\{k\}}\left(\mathbf{s}_{\{k\}}\right) \\ & - \gamma \widehat{V}_{\{k+1,l\}}\left(\mathbf{x}_{C\{k+1\}}, \mathbf{W}_{C\{k,l\}}\right), \end{aligned} \tag{15}$$

$$\begin{aligned} e_{D\{k,l\}} = {} & \mathbf{I}_D^T \left\{ \frac{\partial L_{C\{k\}}\left(\mathbf{s}_{\{k\}}\right)}{\partial \mathbf{s}_{\{k\}}} + \left[\frac{\partial \mathbf{u}_{\{k\}}}{\partial \mathbf{s}_{\{k\}}}\right]^T \frac{\partial L_{C\{k\}}\left(\mathbf{s}_{\{k\}}\right)}{\partial \mathbf{u}_{\{k\}}} \right. \\ & + \gamma \left[\frac{\partial \mathbf{s}_{\{k+1\}}}{\partial \mathbf{s}_{\{k\}}} + \left[\frac{\partial \mathbf{u}_{\{k\}}}{\partial \mathbf{s}_{\{k\}}}\right]^T \frac{\partial \mathbf{s}_{\{k+1\}}}{\partial \mathbf{u}_{\{k\}}} \right]^T \\ & \times \frac{\partial \widehat{V}_{\{k+1,l\}}\left(\mathbf{x}_{C\{k+1\}}, \mathbf{W}_{C\{k,l\}}\right)}{\partial \mathbf{s}_{\{k+1\}}} \\ & \left. - \frac{\partial \widehat{V}_{\{k,l\}}\left(\mathbf{x}_{C\{k\}}, \mathbf{W}_{C\{k,l\}}\right)}{\partial \mathbf{s}_{\{k\}}} \right\}, \end{aligned} \tag{16}$$

where $\mathbf{I}_D$ is a constant vector, $\mathbf{I}_D = [1, 1]^T$. Weights of the critic's NN are adapted using the gradient method according to the equation

$$\begin{aligned} \mathbf{W}_{C\{k,l+1\}} = {} & \mathbf{W}_{C\{k,l\}} - \eta_1 e_{H\{k,l\}} \boldsymbol{\Gamma}_C \frac{\partial \widehat{V}_{\{k,l\}}\left(\mathbf{x}_{C\{k\}}, \mathbf{W}_{C\{k,l\}}\right)}{\partial \mathbf{W}_{C\{k,l\}}} \\ & - \eta_2 e_{D\{k,l\}} \boldsymbol{\Gamma}_C \frac{\partial^2 \widehat{V}_{\{k,l\}}\left(\mathbf{x}_{C\{k\}}, \mathbf{W}_{C\{k,l\}}\right)}{\partial \mathbf{s}_{\{k\}} \mathbf{W}_{C\{k,l\}}}, \end{aligned} \tag{17}$$

where $\boldsymbol{\Gamma}_C$ is the fixed diagonal matrix of positive learning rates and η_1, η_2 are positive constants.

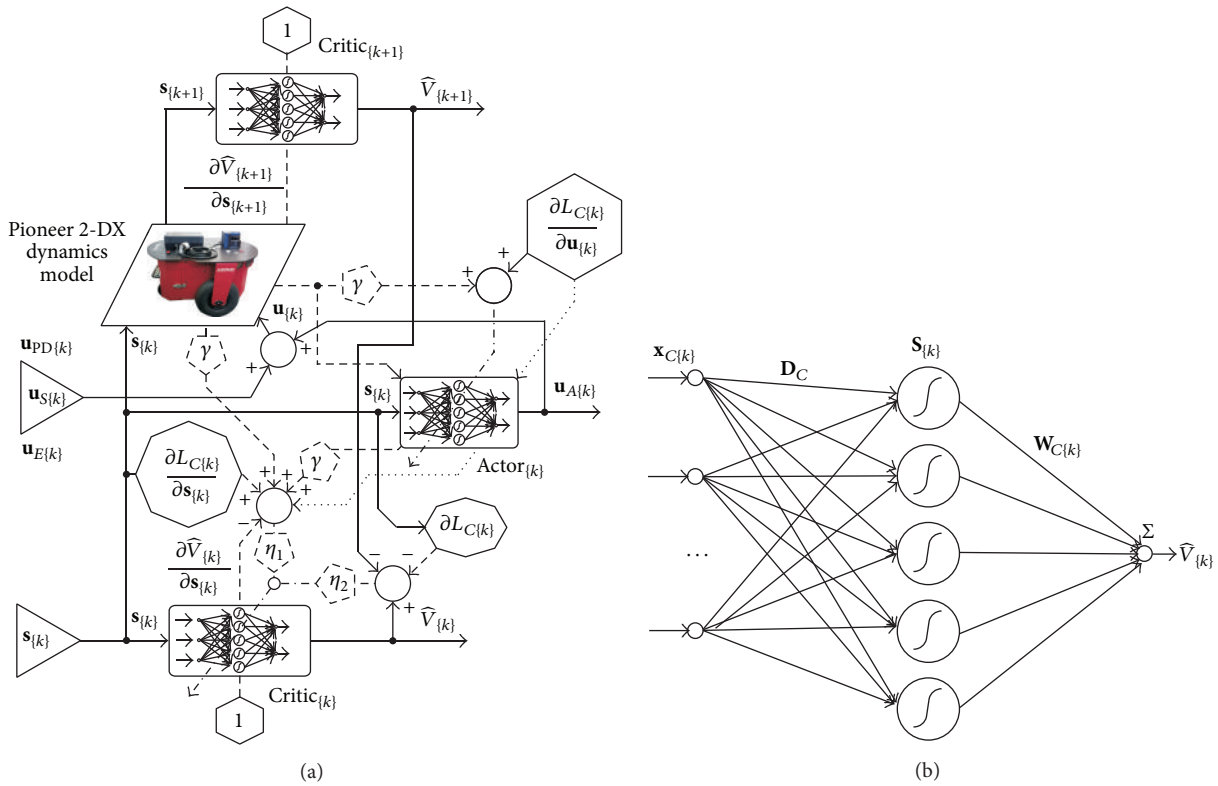


Figure 4: (a) Scheme of the GDHP algorithm, (b) scheme of the critic's RVFL NN.

Adaptation process of NNs' weights is an interesting feature of the ADP algorithms. It is realised in a form of an internal loop with the iteration index l. In every step k of the discrete control process calculations, which are connected to the actor's and the critic's weights adaptation procedure, are executed according to the scheme shown in Figure 5.

The actor-critic structure adaptation process is organised in the following way: at the beginning of every kth iteration step $l = 0$. Actor's NNs weights are adapted according to the assumed adaptation law (12) by minimisation of the error rate (13). This part of the algorithm, called the "control law improvement routine" [9], leads to the evaluation of the actor's NNs weights $\mathbf{W}_{Aj\{k,l+1\}}$. The next step consists of the adaptation of the critic's NN weights; it is called the "value function determination operation." The critic's NN weights are adapted according to the assumed adaptation law, by the minimisation of the error rate (15), called the temporal difference error (TDE) [12], and the error rate (16). This leads to the calculation of the critic's NN weights $\mathbf{W}_{C\{k,l+1\}}$. Next, the internal loop iteration index l is increased, and a new cycle of the ADP algorithm adaptation is started. In the presented algorithm, the internal loop breaks, when the number of internal iterations $l \geq l_{mx}$, where l_{mx} is the maximal number of iteration cycles, or when the error $e_{A[j]\{k,l\}}$ is smaller than the assumed positive limit $E_{A[j]}$, $e_{A[j]\{k,l\}} < E_{A[j]}$, $j = 1, 2$. When one of these conditions is satisfied, $\mathbf{W}_{Aj\{k,l+1\}}$ becomes $\mathbf{W}_{Aj\{k+1,l\}}$ and $\mathbf{W}_{C\{k,l+1\}}$ becomes $\mathbf{W}_{C\{k+1,l\}}$. Next index k is increased. The actor's NNs generate control signals and the GDHP structure receives information about a new state of the controlled object. In the next sections index l is omitted for the sake of simplicity.

5. Stability Analysis

This paper focuses on the implementation of the ADP algorithm in the network-based tracking control system of the two-wheeled mobile robot, Pioneer 2-DX. The proposed discrete tracking control system consists of the GDHP algorithm, the PD controller, the supervisory term, and the additional control signal.

The filtered tracking error $\mathbf{s}_{\{k\}}$ was defined in the form (5), where $\mathbf{\Lambda}$ is a positive defined, fixed diagonal matrix selected in the way that the eigenvalues are within a unit disc. Consequently, if the filtered tracking error (5) tends to zero then all the tracking errors go to zero. Filtered tracking error $\mathbf{s}_{\{k+1\}}$ can be expressed as (6), where the vector $\mathbf{Y}_f(\mathbf{z}_{2\{k\}})$ includes all nonlinearities of the controlled object.

Let us define the control input $\mathbf{u}_{\{k\}}$ as

$$\mathbf{u}_{\{k\}} = h^{-1}\mathbf{M}\left[\mathbf{Y}_d\left(\mathbf{z}_{\{k\}}, \mathbf{z}_{d\{k\}}, \mathbf{z}_{d3\{k\}}\right) - \widehat{\mathbf{Y}}_f\left(\mathbf{z}_{2\{k\}}\right) - \mathbf{K}_D\mathbf{s}_{\{k\}}\right], \tag{18}$$

where $\widehat{\mathbf{Y}}_f(\mathbf{z}_{2\{k\}})$ is an estimate of the unknown function.

Then, the closed-loop system becomes

$$\mathbf{s}_{\{k+1\}} = \mathbf{K}_D\mathbf{s}_{\{k\}} - \widetilde{\mathbf{Y}}_f\left(\mathbf{z}_{2\{k\}}\right) - \mathbf{Y}_{\tau\{k\}}, \tag{19}$$

where the functional estimation error is given by $\widetilde{\mathbf{Y}}_f(\mathbf{z}_{2\{k\}}) = \widehat{\mathbf{Y}}_f(\mathbf{z}_{2\{k\}}) - \mathbf{Y}_f(\mathbf{z}_{2\{k\}})$. Equation (19) relates the filtered tracking

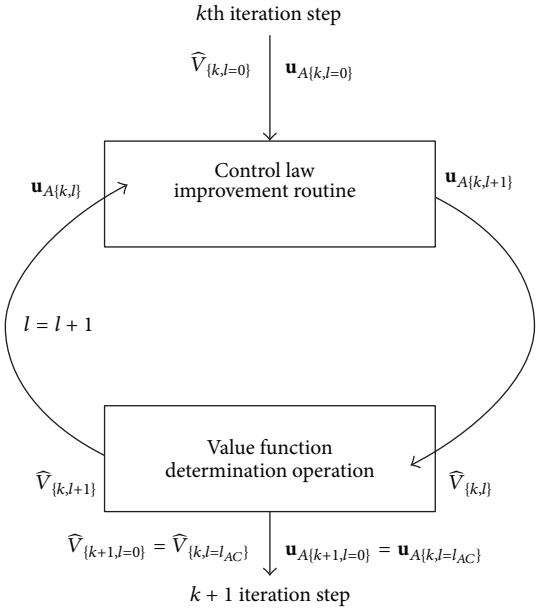


Figure 5: Schematic conception of the ADP structure adaptation process.

error with the functional estimation error. In general, the filtered tracking error system (19) can also be expressed as

$$\mathbf{s}_{\{k+1\}} = \mathbf{K}_D \mathbf{s}_{\{k\}} + \mathbf{d}_{0\{k\}}, \tag{20}$$

where $\mathbf{d}_{0\{k\}} = -(\widetilde{\mathbf{Y}}_f(\mathbf{z}_{2\{k\}}) + \mathbf{Y}_{\tau\{k\}})$. If the functional estimation error $\widetilde{\mathbf{Y}}_f(\mathbf{z}_{2\{k\}})$ is bounded in such a way that $|Y_{f[j]}(\mathbf{z}_{2\{k\}})| \leq F_{[j]}$, $F_{[j]}$ is a positive constant and $Y_{\tau[j]\{k\}} < b_{d[j]}$, where $b_{d[j]}$ is a positive constant, then the next stability results hold.

Let us consider the system given by (3). Let the control action be provided by (18) and assume that the functional estimation error and the unknown disturbance are bounded. The filtered tracking error system (6) is stable provided that

$$0 < K_{D\max} < 1, \tag{21}$$

where $K_{D\max} \in R$ is the maximum eigenvalue of the matrix $\mathbf{K}_D$.

Let us consider the following Lyapunov function candidate:

$$L_{\{k\}} = \mathbf{s}_{\{k\}}^T \mathbf{s}_{\{k\}}. \tag{22}$$

The first difference is

$$\Delta L_{\{k\}} = \mathbf{s}_{\{k+1\}}^T \mathbf{s}_{\{k+1\}} - \mathbf{s}_{\{k\}}^T \mathbf{s}_{\{k\}}. \tag{23}$$

Substituting the filtered tracking error dynamics (19) into (23) results in

$$\begin{aligned} \Delta L_{\{k\}} = {} & \left[\mathbf{K}_D \mathbf{s}_{\{k\}} - \widetilde{\mathbf{Y}}_f\left(\mathbf{z}_{2\{k\}}\right) - \mathbf{Y}_{\tau\{k\}}\right]^{\mathbf{T}} \\ & \times \left[\mathbf{K}_D \mathbf{s}_{\{k\}} - \widetilde{\mathbf{Y}}_f\left(\mathbf{z}_{2\{k\}}\right) - \mathbf{Y}_{\tau\{k\}}\right] - \mathbf{s}_{\{k\}}^T \mathbf{s}_{\{k\}}, \end{aligned} \tag{24}$$

what implies that $\Delta L_{\{k\}} \leq 0$ provided that

$$\left\|\mathbf{K}_D \mathbf{s}_{\{k\}} - \widetilde{\mathbf{Y}}_f\left(\mathbf{z}_{2\{k\}}\right) - \mathbf{Y}_{\tau\{k\}}\right\| \leq K_{D\max} \left\|\mathbf{s}_{\{k\}}\right\| + F + b_d < \left\|\mathbf{s}_{\{k\}}\right\|, \tag{25}$$

where F and b_d are positive constants. This further implies that

$$\left\|\mathbf{s}_{\{k\}}\right\| \geq \frac{F + b_d}{1 - K_{D\max}}. \tag{26}$$

The closed-loop system is uniformly ultimately bounded (UUB) [47]. The PD controller parameter $K_{D\max} \in R$ has to be selected using (21) in order for the closed-loop system to be stable. This outer-loop signal is viewed as the supervisor's evaluation feedback to the actor and the critic. In the NN actor-critic control scheme derived in this paper there is no preliminary offline learning phase. The weights are simply initialized at zero, for then the control system is just the PD controller. Therefore, the closed-loop system remains stable until the NNs begin to learn.

The proposed discrete tracking control system is composed of the GDHP structure that generates the control signal $\mathbf{u}_{A\{k\}}$, the PD controller ($\mathbf{u}_{\mathrm{PD}\{k\}}$), the supervisory term ($\mathbf{u}_{S\{k\}}$), and the additional control signal $\mathbf{u}_{E\{k\}}$. Structure of the supervisory term derives from the stability analysis performed using the Lyapunov stability theorem. The additional control signal $\mathbf{u}_{E\{k\}}$ derives from the process of the WMR dynamics model discretisation. The overall tracking control signal was assumed in the form

$$\mathbf{u}_{\{k\}} = -h^{-1}\mathbf{M}\left[\mathbf{u}_{A\{k\}} + \mathbf{u}_{\mathrm{PD}\{k\}} + \mathbf{u}_{E\{k\}} - \mathbf{u}_{S\{k\}}\right], \tag{27}$$

where

$$\begin{aligned} \mathbf{u}_{\mathrm{PD}\{k\}} &= \mathbf{K}_D \mathbf{s}_{\{k\}}, \\ \mathbf{u}_{E\{k\}} &= h\left[\mathbf{\Lambda} \mathbf{e}_{2\{k\}} - \mathbf{z}_{d3\{k\}}\right], \\ \mathbf{u}_{S\{k\}} &= \mathbf{I}_S \mathbf{u}_{S\{k\}}^*, \end{aligned} \tag{28}$$

where $\mathbf{K}_D$ is a fixed diagonal matrix of positive PD controller gains, $\mathbf{I}_S$ is a diagonal matrix, with elements $I_{S[j,j]} = 1$ if $|s_{[j]\{k\}}| \geq \rho_{[j]}$ or $I_{S[j,j]} = 0$ in the other case, $j = 1, 2$, $\rho_{[j]}$ is a positive constant.

The scheme of the discrete neural tracking control system with actor-critic structure in the GDHP configuration is shown in Figure 6.

The stability analysis was performed under the assumption that $I_{S[j,j]} = 1$. Substituting (27) into (6), the closed-loop system equation is expressed by the formula

$$\begin{aligned} \mathbf{s}_{\{k+1\}} = {} & \mathbf{Y}_d\left(\mathbf{z}_{\{k\}}, \mathbf{z}_{d\{k\}}, \mathbf{z}_{d3\{k\}}\right) - \mathbf{Y}_f\left(\mathbf{z}_{2\{k\}}\right) - \mathbf{Y}_{\tau\{k\}} \\ & - \left[\mathbf{u}_{A\{k\}} + \mathbf{u}_{\mathrm{PD}\{k\}} + \mathbf{u}_{E\{k\}} - \mathbf{u}_{S\{k\}}\right]. \end{aligned} \tag{29}$$

The stability analysis was realised using the positive definite Lyapunov candidate function

$$L = \frac{1}{2}\mathbf{s}^T \mathbf{s}, \tag{30}$$

which discretised derivative was assumed in the form

$$\Delta L_{\{k\}} = \mathbf{s}_{\{k\}}^T \left[\mathbf{s}_{\{k+1\}} - \mathbf{s}_{\{k\}}\right]. \tag{31}$$

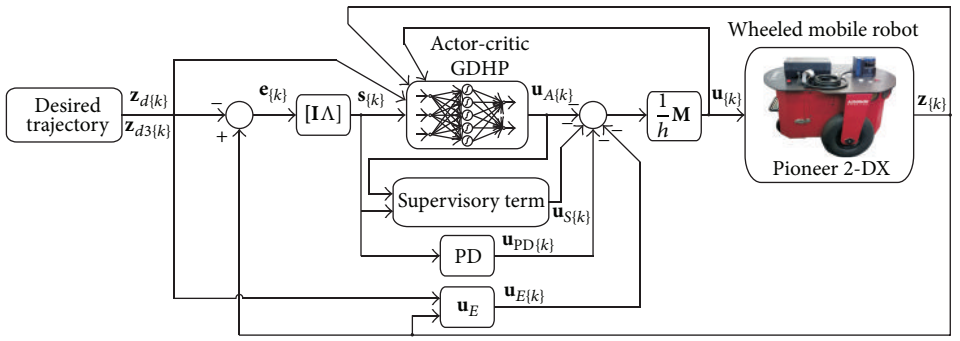


Figure 6: Scheme of the tracking control system.

Substituting (29) into (31), $\Delta L_{\{k\}}$ takes the form

$$\Delta L_{\{k\}} = \mathbf{s}_{\{k\}}^T \left[-\mathbf{Y}_f\left(\mathbf{z}_{2\{k\}}\right) - \mathbf{Y}_{\tau\{k\}} - \mathbf{u}_{A\{k\}} - \mathbf{K}_D\mathbf{s}_{\{k\}} + \mathbf{u}_{S\{k\}}^*\right]. \tag{32}$$

On the assumption that all elements of the vector of disturbances are bounded, $Y_{\tau[j]\{k\}} < b_{d[j]}$, where $b_{d[j]}$ is a positive constant, the difference of the Lyapunov candidate function takes the form

$$\begin{aligned}\Delta L_{\{k\}} \le & -\mathbf{s}_{\{k\}}^T\mathbf{K}_D\mathbf{s}_{\{k\}} + \sum_{j=1}^{2}\left|s_{[j]\{k\}}\right| \\ & \times\left[\left|Y_{f[j]}\left(\mathbf{z}_{2\{k\}}\right)\right| + \left|u_{A[j]\{k\}}\right| + \left|Y_{\tau[j]\{k\}}\right|\right] \\ & + \sum_{j=1}^{2} s_{[j]\{k\}}u_{S[j]\{k\}}^*.\end{aligned} \tag{33}$$

The supervisory term's control signal was assumed in the form

$$u_{S[j]\{k\}}^* = -\operatorname{sgn}\left(s_{[j]\{k\}}\right)\left[F_{[j]} + \left|u_{A[j]\{k\}}\right| + b_{d[j]} + \sigma_{[j]}\right], \tag{34}$$

where $|Y_{f[j]}(\mathbf{z}_{2\{k\}})| \le F_{[j]}$, $F_{[j]}$ is a positive constant, and $\sigma_{[j]}$ is a positive constant. On the above assumptions the difference of the Lyapunov function (30) is a negative definite.

6. Research Results

Performance of the proposed discrete tracking control system was tested during a series of computer simulations and then verified using the laboratory stand schematically shown in Figure 7.

The laboratory stand consists of the WMR Pioneer 2-DX, the power supply and a PC equipped with the dSpace DS1102 digital signal processing board and software: dSpace Control Desk and Matlab/Simulink. The WMR Pioneer 2-DX is equipped with the sensory system composed of eight ultrasonic sensors and a scanning laser range finder. The movement of the robot is realised using two independently supplied DC motors with gears (ratio 19.7 : 1) and encoders (500 ticks per shaft revolution). The WMR weights $m_R = 9$ kg, its frame is $l_R = 0.44$ m long, $l_W = 0.33$ m width, and its maximal velocity is equal to $v_A = 1.6$ m/s.

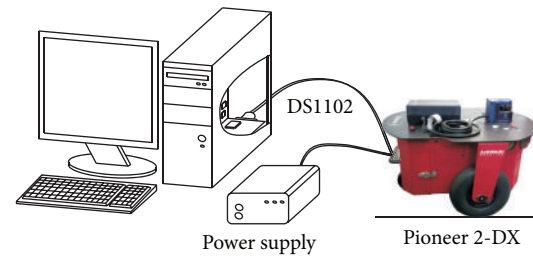


Figure 7: Scheme of the laboratory stand.

6.1. Simulation Results. Performance of the proposed control system was tested during a series of numerical simulations performed using the Matlab/Simulink software environment. In this section the notation of variables is simplified and the index k is omitted. The same set of parameters during simulations as in the experiment was used. The time discretisation parameter was equal to $h = 0.01$ s. In the GDHP structure NNs with eight neurons each were used. The output layer weights of NNs were set to zero in the initialisation process. Parameters of the PD controller $\mathbf{K}_D = \operatorname{diag}\{0.036, 0.036\}$, $\mathbf{\Lambda} = \operatorname{diag}\{0.5, 0.5\}$ were assumed. One must select $\mathbf{K}_D$ using some trial and error experiments or computer simulations. In practice, this has not shown itself to be a problem. The PD controller gains were selected heuristically to satisfy (21). For the sake of the noise that occurs in the signals of the driving wheels angular velocities, incremental encoders were used in the experiment for measurement, the amplification of PD gains in a range of conditions (21) does not improve tracking control quality and can lead to instability. The matrix $\mathbf{R}$, in the cost function, was set to $\mathbf{R} = \operatorname{diag}\{1, 1\}$, the discount factor was equal to $\gamma = 0.5$, learning rates of the actor's NNs and the critic's NN were equal to $\Gamma_{A[i,i]} = 0.1$ and $\Gamma_{C[i,i]} = 0.9$ properly, $i = 1, \ldots, 8$, $\eta_1 = \eta_2 = 1$. Parameters of the supervisory term were set to $\rho_{[j]} = 3$ and $\sigma_{[j]} = 0.09$. The maximal velocity of point A of the WMR's frame was equal to $v_A = 0.4$ m/s. During the movement of the WMR two parametric disturbances were simulated (marked on diagrams by ellipses), first in $t_1 = 12.5$ s, when the nominal set of parameters was changed to $\mathbf{a}_d = [0.1343, 0.0945, 0.037, 0.0001, 2.296, 2.296]^T$ and the second one, when in $t_1 = 32.5$ s, nominal

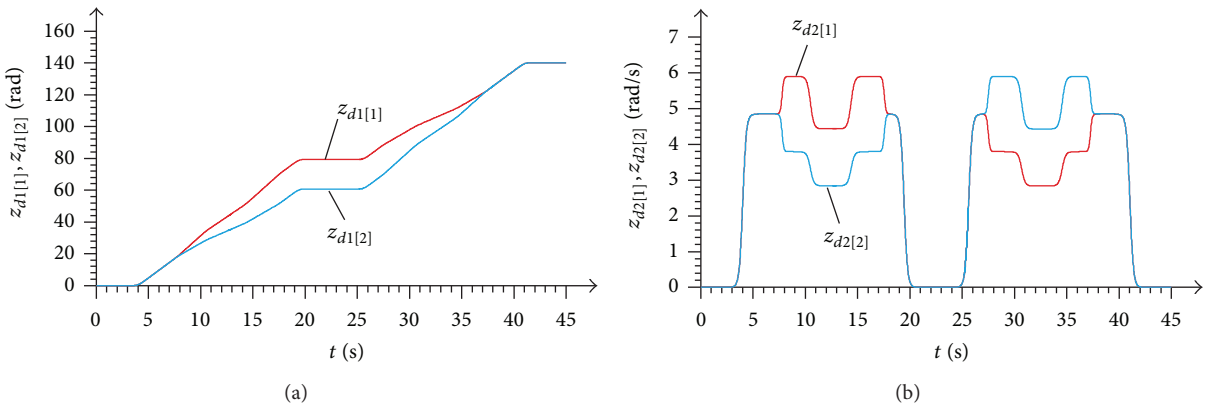


FIGURE 8: (a) The desired angles of wheels 1 and 2 rotation, $z_{d1[1]}$ and $z_{d1[2]}$, (b) the desired angular velocities of driving wheels 1 and 2, $z_{d2[1]}$ and $z_{d2[2]}$.

values of parameters were restored. The first change of parameters corresponds to the situation, when the WMR is loaded by an additional mass $m_L = 5$ kg, and a return to the nominal set of parameters corresponds to the situation, when the additional load is removed.

The desired trajectory of the WMR was computed earlier. In Figure 8(a) the desired angles of the driving wheels', 1 and 2, rotation are shown; in Figure 8(b) the desired angular velocities are presented. Realisation of the presented trajectory results in movement of point A of the WMR on the path in a shape of a digit "8," with a stop phase in the middle point.

The overall tracking control signal $\mathbf{u}$, shown in Figure 9(a), consists of the control signals generated by the actor's NNs $\mathbf{u}_A$, (Figure 9(b)), the PD control signals $\mathbf{u}_{\mathrm{PD}}$, (Figure 9(c)), the supervisory term's control signals $\mathbf{u}_S$, and the additional control signals $\mathbf{u}_E$, shown both in Figure 9(d). At the beginning of the numerical test, values of the PD control signals are big. Next, they are reduced during the NNs adaptation process. The control signals of the actor take the main part in the overall control signals. In time t_1, when the first parametric disturbance occurs, a change in values of the generated control signals can be observed. The additional load changes the dynamics of the WMR; realisation of the desired trajectory requires generating higher values of the control signals. The influence of the disturbance on the WMR's dynamics is compensated by the actor's NNs control signals. Analogically, the change of the WMR's parameters in time t_2, which simulates removal of the additional load, is compensated in the generated control law by reduction of the actor's NNs control signals values.

The desired and realised angular velocities of driving wheels 1 and 2 are shown in Figures 10(a) and 10(b), respectively. The biggest differences between the desired and realised angular velocities occur at the beginning of the numerical test. Small changes of realised angular velocities can be observed at the moment, when the parametric disturbances occur.

The desired trajectory was realised with tracking errors shown in Figures 11(a) and 11(b) for adequate driving wheels. In Figures 11(c) and 11(d), values of filtered tracking errors $s_{[1]}$ and $s_{[2]}$ are shown that are minimised by the ADP structure. The highest values of the tracking errors occur at the beginning of the numerical test, when values of the PD control signals are at their highest, and the process of NNs' zero initial weights adaptation starts. Next, the control signals of the actor's NNs take the main part of the overall control signals, and the values of tracking errors are reduced. A noticeable increase of the tracking error values occurs at the time of simulated disturbances, but it is reduced by the change of the actor's NNs control signals.

Values of the GDHP structure's NNs weights are shown in Figure 12(a) for the first actor's NN, in Figure 12(b) for the second one, and in Figure 12(c) for the critic's NN. In the numerical test, zero initial weights values were used. At the time of the disturbances, changes of weights' values occur as a result of the adaptation performed in order to reduce the tracking errors.

6.2. Verification Results. After numerical tests were performed, a series of experiments were realised using the WMR Pioneer 2-DX. The control algorithm operated in real time during the experiment, thanks to the application of the dSpace DS1102 digital signal processing board. In the experiment, the same parameters of the control system as in the simulation were used. The values of signals from the experiment were not filtered. The control signals are shown in Figure 13. The first disturbance occurs at time $t_1 = 13$ s and the second one at time $t_2 = 33$ s. The PD control signals (Figure 13(c)) based on the tracking errors calculated on the basis of the realised trajectory, determined by using signals form incremental encoders. These signals are noised, which has an effect on the overall control signals (Figure 13(a)). In contrast, the actor's NNs control signals (Figure 13(b)) and residual control signals (Figure 13(d)) are smooth. As it was observed in the simulation, at the time of the disturbances, the values of the actor's NNs control signals changed to compensate the effect of the WMR's dynamics change.

The biggest differences between the desired and realised angular velocities, shown in Figure 14, occur at the beginning of the experiment, when the process of the actor's NNs weights adaptation starts and at the time when the disturbances occur.

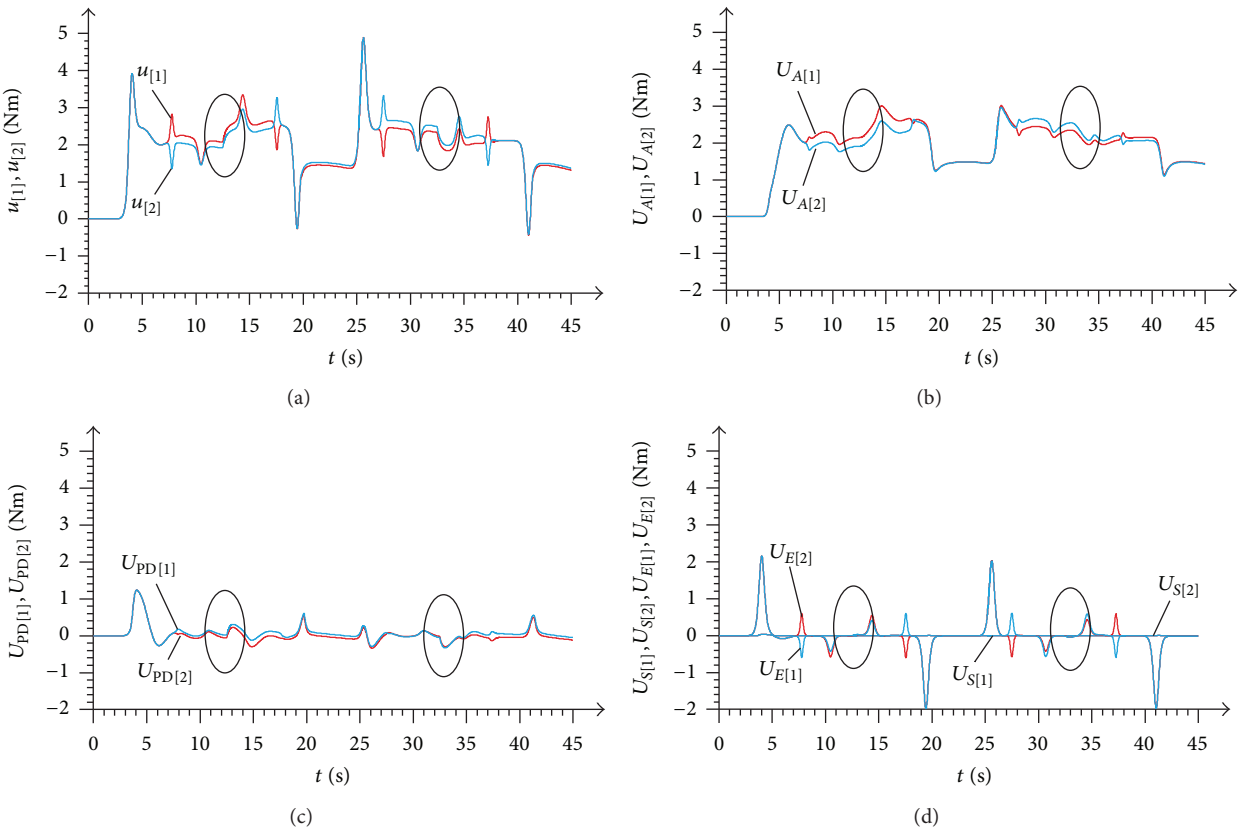


FIGURE 9: (a) The overall tracking control signals $u_{[1]}$ and $u_{[2]}$, (b) the actor's NNs control signals $U_{A[1]}$ and $U_{A[2]}$, $\mathbf{U}_A = -h^{-1}\mathbf{M}\mathbf{u}_A$, (c) the PD control signals $U_{PD[1]}$ and $U_{PD[2]}$, $\mathbf{U}_{PD} = -h^{-1}\mathbf{M}\mathbf{u}_{PD}$, (d) the supervisory term's control signals ($U_{S[1]}$, $U_{S[2]}$), $\mathbf{U}_S = -h^{-1}\mathbf{M}\mathbf{u}_S$, and the control signals $U_{E[1]}$ and $U_{E[2]}$, $\mathbf{U}_E = -h^{-1}\mathbf{M}\mathbf{u}_E$.

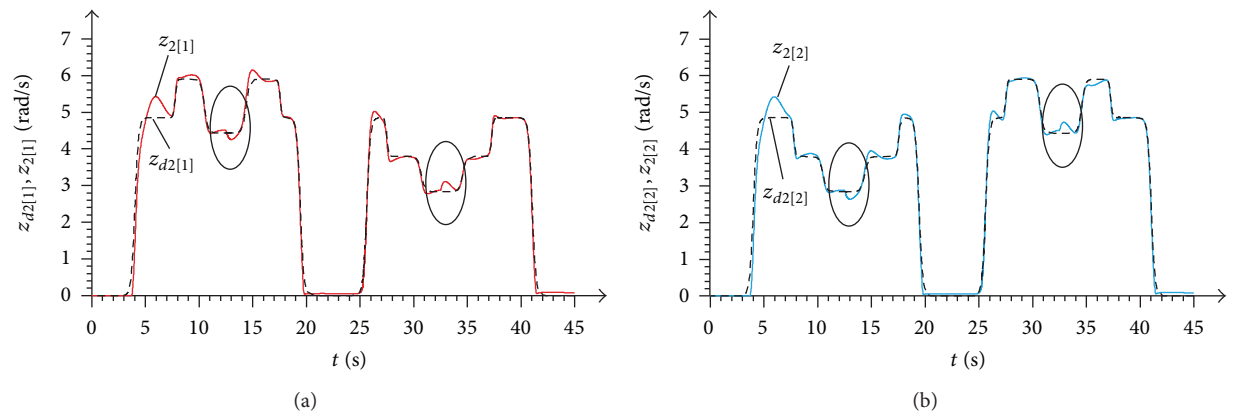


FIGURE 10: (a) The desired (dashed line) and realised (continuous line) angular velocity of wheel 1, $z_{d2[1]}$ and $z_{2[1]}$, (b) the desired (dashed line) and realised (continuous line) angular velocity of wheel 2, $z_{d2[2]}$ and $z_{2[2]}$.

The tracking errors of wheels 1 and 2 are shown in Figures 15(a) and 15(b); filtered tracking errors are shown in Figures 15(c) and 15(d). Values of errors are noisy, because of the realised method of measurement of the movement parameters. The errors at the beginning of the experiment are at their highest. The change of the load transported by the WMR has noticeable influence on the trajectory realisation process. The method of placing the load on the WMR and removing it has a big influence on temporary values of errors. The increase of errors values results in the adaptation of the actor's and the critic's NNs weights in order to minimise tracking errors.

Values of NNs' weights are shown in Figure 16. At a time, when the WMR transports an additional load, values of the

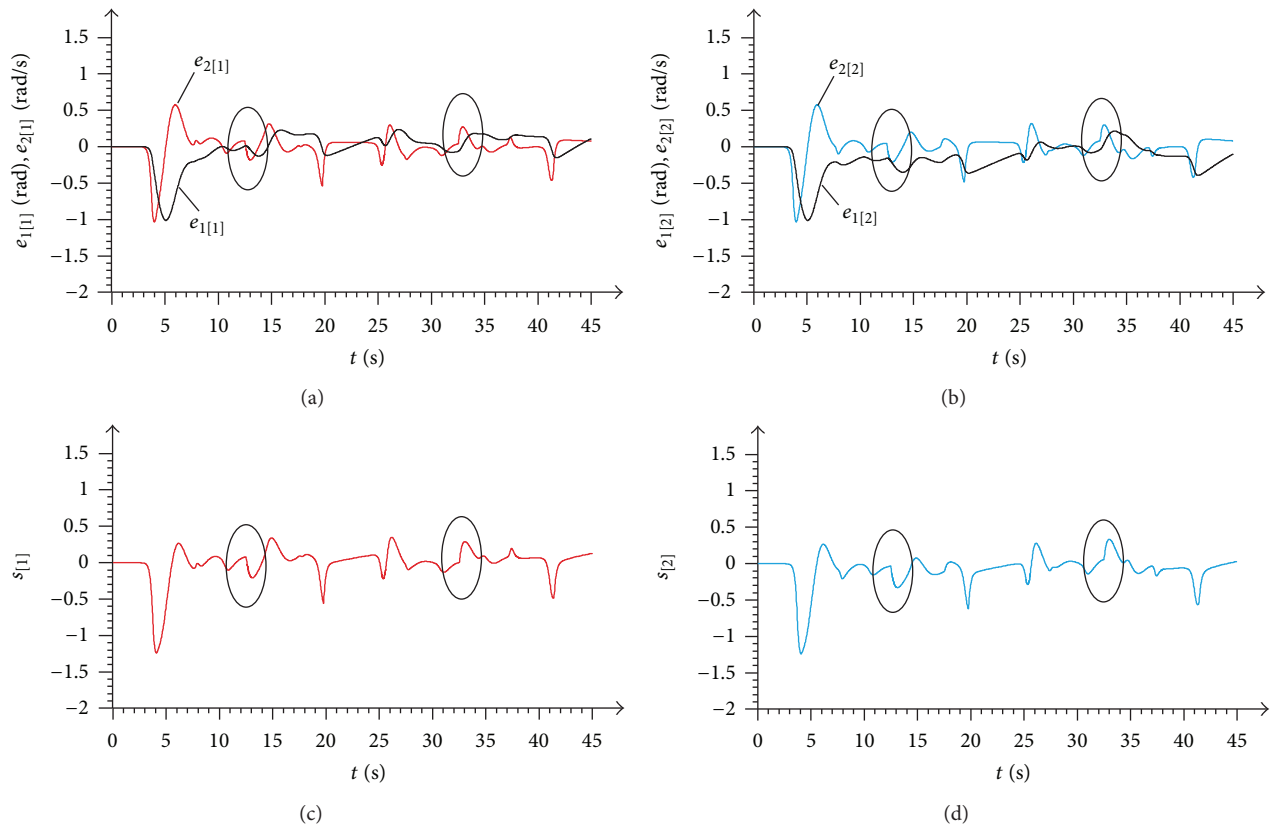


FIGURE 11: (a) Tracking errors of wheel 1, $e_{1[1]}$ and $e_{2[1]}$, (b) tracking errors of wheel 2, $e_{1[2]}$ and $e_{2[2]}$, (c) the filtered tracking error $s_{[1]}$, and (d) the filtered tracking error $s_{[2]}$.

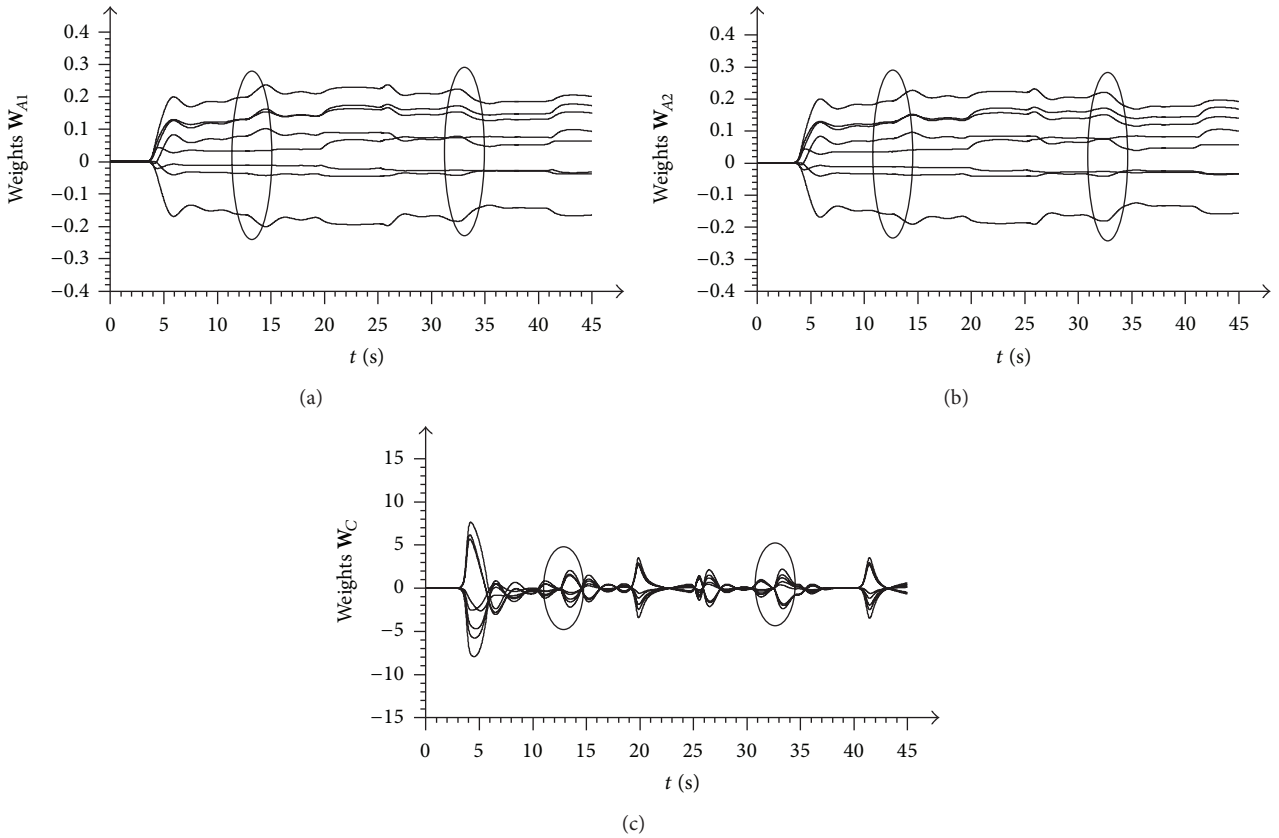


FIGURE 12: (a) Weights of the actor's 1 RVFL NN $\mathbf{W}_{A1}$, (b) weights of the actor's 2 RVFL NN $\mathbf{W}_{A2}$, and (c) weights of the critic's RVFL NN $\mathbf{W}_C$.

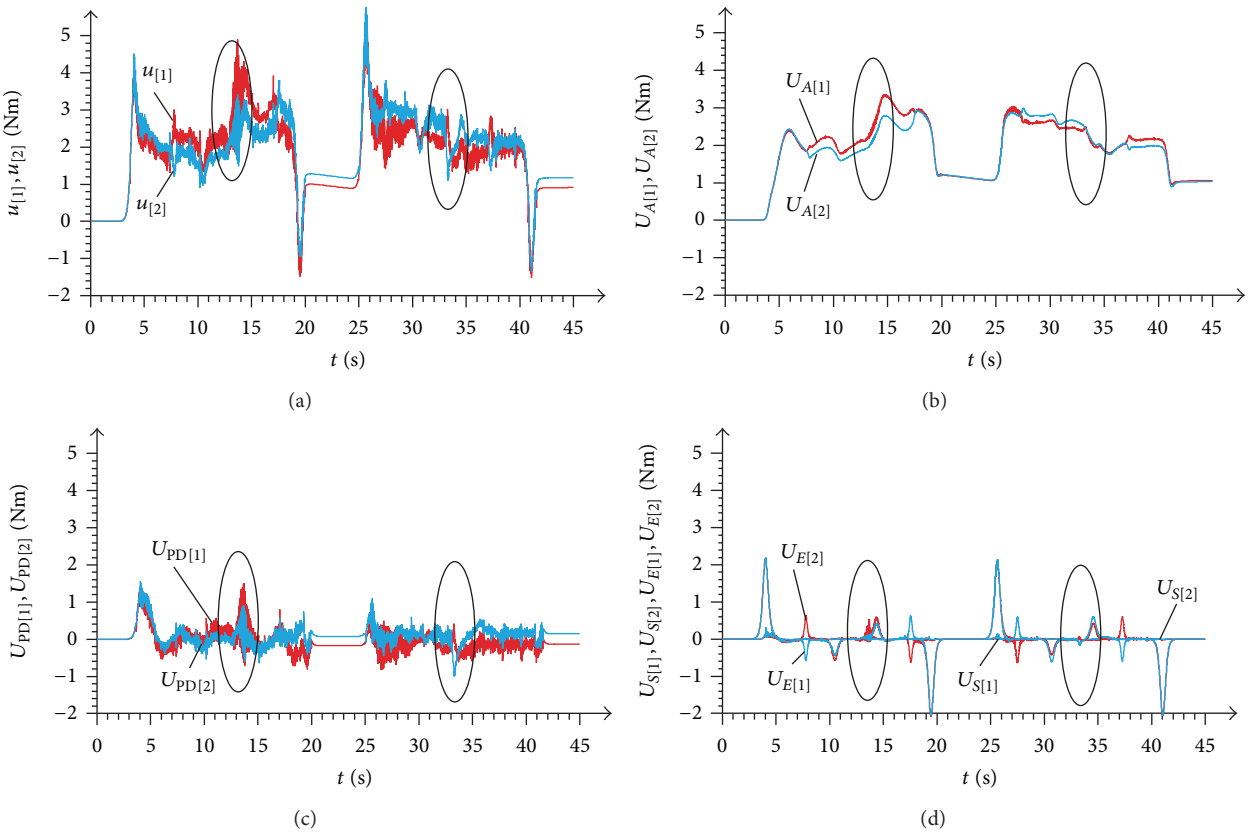


FIGURE 13: (a) The overall tracking control signals $u_{[1]}$ and $u_{[2]}$, (b) the actor's NNs control signals $U_{A[1]}$ and $U_{A[2]}$, $\mathbf{U}_A = -h^{-1}\mathbf{M}\mathbf{u}_A$, (c) the PD control signals $U_{PD[1]}$ and $U_{PD[2]}$, $\mathbf{U}_{PD} = -h^{-1}\mathbf{M}\mathbf{u}_{PD}$, (d) the supervisory term's control signals $(U_{S[1]}, U_{S[2]})$, $\mathbf{U}_S = -h^{-1}\mathbf{M}\mathbf{u}_S$, and the control signals $U_{E[1]}$ and $U_{E[2]}$, $\mathbf{U}_E = -h^{-1}\mathbf{M}\mathbf{u}_E$.

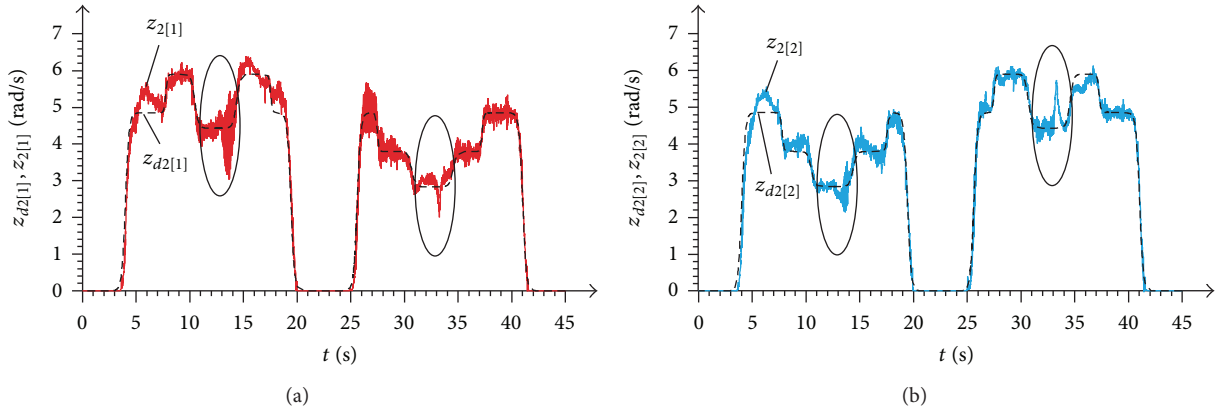


FIGURE 14: (a) The desired (dashed line) and realised (continuous line) angular velocity of wheel 1, $z_{d2[1]}$ and $z_{2[1]}$, (b) the desired (dashed line) and realised (continuous line) angular velocity of wheel 2, $z_{d2[2]}$ and $z_{2[2]}$.

actor's NNs weights increase. This is a result of generating higher values of the actor's control signals for the heavier WMR. The critic's NN approximates the value function based on the filtered tracking errors, values of its weights increase and when the values of filtered tracking errors increase.

The tracking quality of the proposed control system was compared to the results obtained by the tracking control systems presented earlier, where ADP algorithms in HDP and DHP [43] configuration, or the PD controller ($\mathbf{K}_D = \text{diag}\{1, 1\}$, $\mathbf{\Lambda} = \text{diag}\{0.5, 0.5\}$), were used. Every experiment was performed in the same conditions, using the same or

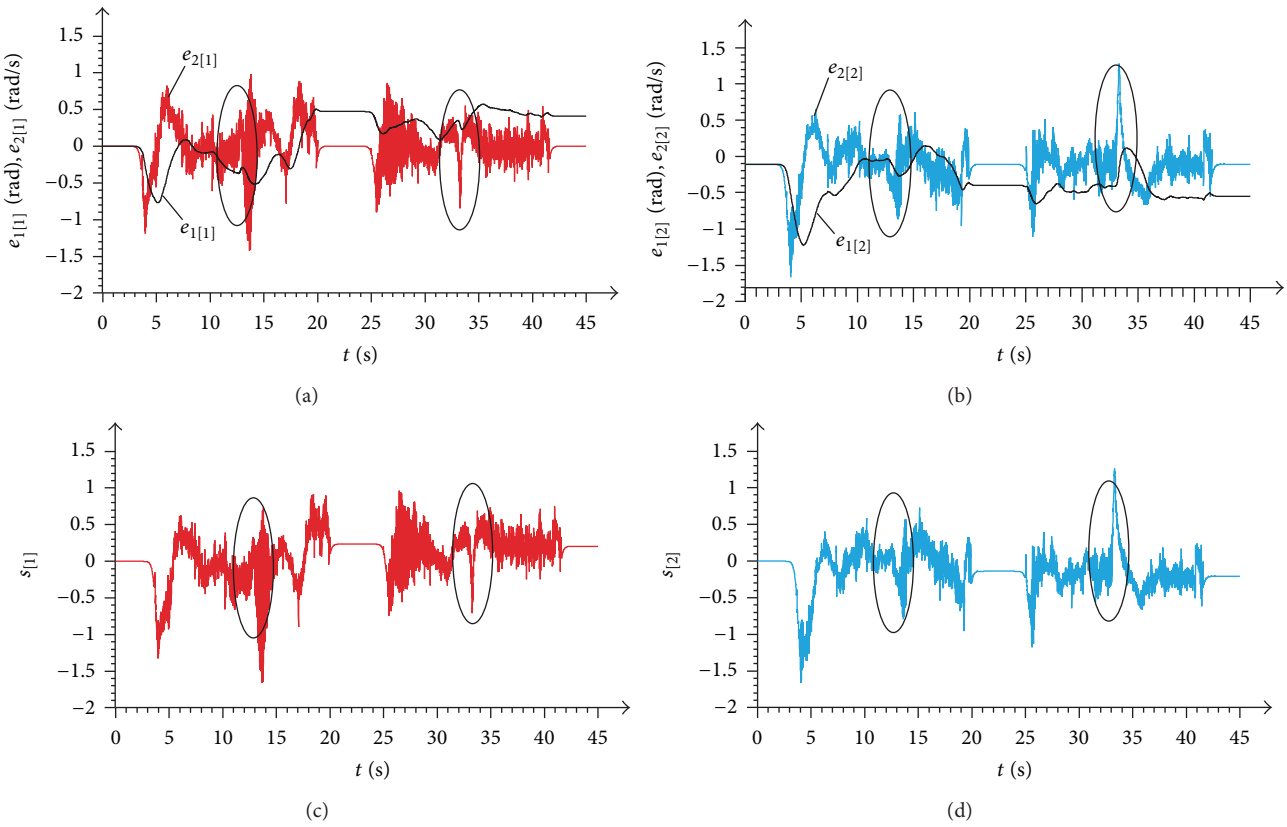


FIGURE 15: (a) Tracking errors of wheel 1, $e_{1[1]}$ and $e_{2[1]}$, (b) tracking errors of wheel 2, $e_{1[2]}$ and $e_{2[2]}$, (c) the filtered tracking error $s_{[1]}$, and (d) the filtered tracking error $s_{[2]}$.

analogical values of parameters, and the same type of the disturbance.

To evaluate the tracking control quality, the following quality ratings were used:

(i) average of maximal values of the filtered tracking error for wheels 1 ($s_{\max[1]}$) and 2 ($s_{\max[2]}$):

$$s_{\text{mavr}} = \frac{1}{2}\left(s_{\max[1]} + s_{\max[2]}\right), \tag{35}$$

(ii) average of root mean square error (RMSE) of the filtered tracking errors $s_{[1]}$ and $s_{[2]}$:

$$\varepsilon_{\text{avr}} = \frac{1}{2}\left(\sqrt{\frac{1}{N}\sum_{k=0}^{N} s_{[1]\{k\}}^2} + \sqrt{\frac{1}{N}\sum_{k=0}^{N} s_{[2]\{k\}}^2}\right), \tag{36}$$

where $N = 4500$.

Values of quality ratings are shown in Table 1.

Average of maximal values of the filtered tracking error for wheels 1 ($s_{\max[1]}$) and 2 ($s_{\max[2]}$) is shown in Figure 17(a), and values of RMSE of the filtered tracking errors $s_{[1]}$ and $s_{[2]}$ are shown in Figure 17(b).

On the basis of the obtained results, the higher quality of tracking for the control systems with ADP algorithms in comparison to the PD controller can be noticed. In the presented paper the goal was not to demonstrate the maximal quality of the tracking control attainable using

TABLE 1: Values of quality ratings.

Control algorithm	PD	HDP	GDHP	DHP
s_{mavr}	4.06	2.24	1.66	1.52
ε_{avr}	1.99	0.42	0.32	0.24

highest feasible to apply the PD controller gains but to illustrate the increase of the quality of the tracking control after adding, to the control system, a part that compensates for nonlinearities of the control system. Values of the quality ratings for the control system with the GDHP structure are close to the ones obtained by the control system with the DHP structure. Simultaneously values of quality ratings are lower than obtained using the HDP algorithm, which means that the application of more complex critic's NN weights adaptation rule improves the quality of control.

7. Conclusion

The paper presents the discrete tracking control system of the WMR Pioneer 2-DX. The main element of the control system is the ADP algorithm in the GDHP configuration. It consists of the actor and the critic, realised in a form of RVFL NNs. The additional elements of the control system, like the PD controller or the supervisory term, assure stability of the tracking control in case of disturbances, or at the

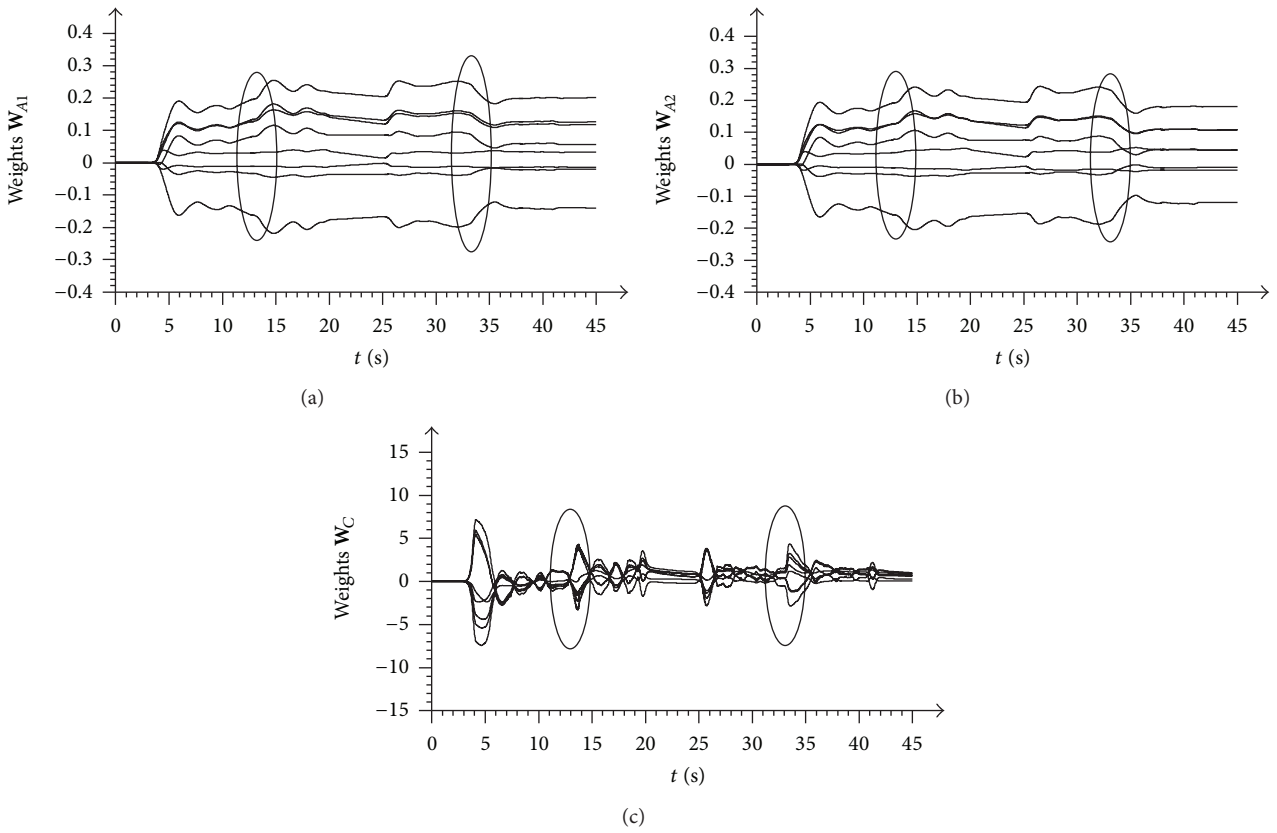


FIGURE 16: (a) Weights of the actor's 1 RVFL NN $\mathbf{W}_{A1}$, (b) weights of the actor's 2 RVFL NN $\mathbf{W}_{A2}$, and (c) weights of the critic's RVFL NN $\mathbf{W}_C$.

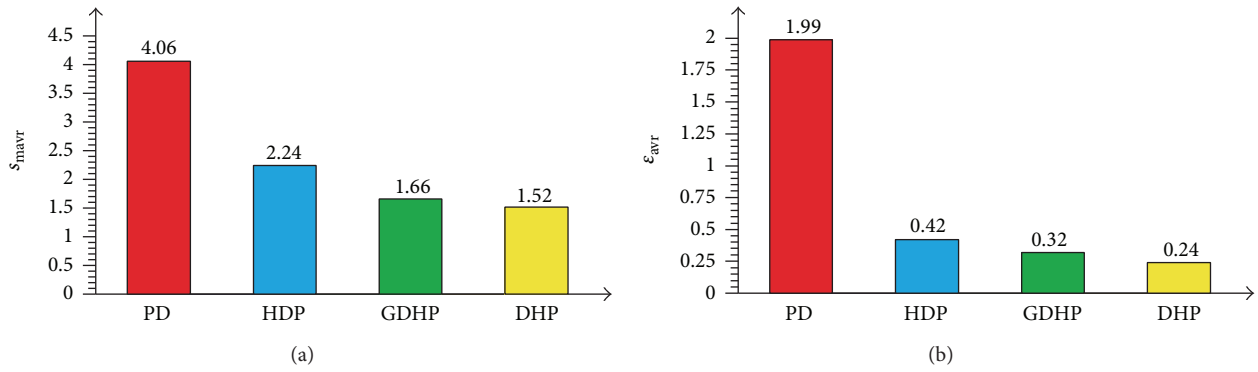


FIGURE 17: (a) Average of maximal values of the filtered tracking error for wheels 1 ($s_{\max[1]}$) and 2 ($s_{\max[2]}$), (b) RMSE of the filtered tracking errors $s_{[1]}$ and $s_{[2]}$.

beginning of movement, in the case when values of the actor's NNs weights are not adequately selected for the controlled system; for example, the process of preliminary learning was not realised, or zero initial weights were applied. PD controller gains were selected experimentally for the control system with the GDHP algorithm. Next the experiment for the control system with only the PD controller, with the same parameters, was performed to demonstrate the increase of the tracking control quality for the tracking control system compensating nonlinearities of the control object. It is important to indicate that in a case of realisation of the control system, with nonlinearities compensation, the primary part of the system is the nonlinear compensator. The nonlinear compensator, realised in the form of a GDHP algorithm, compensates for the nonlinearities of the controlled object, as well as the parametrical and the structural disturbances. The GDHP algorithm has the same structure as HDP and its critic's structure is simpler than in DHP. In the GDHP algorithm the critic's NN weights are adapted using a more complex adaptation law, which is composed of the critic's

NN weights adaptation rule of the HDP algorithm and the DHP algorithm. This feature assures a high quality of tracking, higher than the quality of tracking obtained when using the control system with the HDP algorithm, and close to the quality of tracking for the control system with the DHP algorithm, which is a significant advantage. The presented control system is stable; the values of errors and NNs' weights are bounded. Even in the case of zero initial weights of NNs application, or in the case of disturbances, the proposed control system guarantees a stable tracking process. The discrete tracking control system works online and does not require a process of preliminary learning of NNs. Performance of the control system was verified by a series of numerical tests and experiments realised using the WMR Pioneer 2-DX.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

References

[1] R. Fierro and F. L. Lewis, "Control of a nonholonomic mobile robot using neural networks," *IEEE Transactions on Neural Networks*, vol. 9, no. 4, pp. 589–600, 1998.

[2] M. Giergiel, Z. Hendzel, and W. Zylski, *Modelling and Control of Wheeled Mobile Robots*, PWN, Warsaw, 2002, (Polish).

[3] D. Liu, D. Wang, and X. Yang, "An iterative adaptive dynamic programming algorithm for optimal control of unknown discrete-time nonlinear systems with constrained inputs," *Information Sciences*, vol. 220, pp. 331–342, 2013.

[4] R. Syam, K. Watanabe, and K. Izumi, "Adaptive actor-critic learning for the control of mobile robots by applying predictive models," *Soft Computing*, vol. 9, no. 11, pp. 835–845, 2005.

[5] W. T. Miller, R. S. Sutton, and P. J. Werbos, Eds., *Neural Networks for Control, A Bradford Book*, The MIT Press, Cambridge, Mass, USA, 1990.

[6] Z. Wiesław and G. Piotr, "Verification of multilayer neural-net controller in manipulator tracking control," *Solid State Phenomena*, vol. 164, pp. 99–104, 2010.

[7] R. Bellman, *Dynamic Programming*, Princeton University Press, Princeton, NJ, USA, 1957.

[8] A. G. Barto, R. S. Sutton, and C. W. Anderson, "Neuronlike adaptive elements that can solve difficult learning control problems," *IEEE Transactions on Systems, Man and Cybernetics*, vol. 13, no. 5, pp. 834–846, 1983.

[9] A. G. Barto, W. B. Powell, J. Si, and D. Wunsch, *Handbook of Learning and Approximate Dynamic Programming*, Wiley-IEEE Press, New York, NY, USA, 2004.

[10] A. G. Barto and R. Sutton, *Reinforcement Learning: An Introduction*, MIT Press, Cambridge, Mass, USA, 1998.

[11] W. B. Powell, *Approximate Dynamic Programming: Solving the Curses of Dimensionality*, Wiley, Hoboken, NJ, USA, 2007.

[12] D. V. Prokhorov and D. C. Wunsch II, "Adaptive critic designs," *IEEE Transactions on Neural Networks*, vol. 8, no. 5, pp. 997–1007, 1997.

[13] F.-Y. Wang, H. Zhang, and D. Liu, "Adaptive dynamic programming: an introduction," *IEEE Computational Intelligence Magazine*, vol. 4, no. 2, pp. 39–47, 2009.

[14] F. L. Lewis, D. Liu, and G. G. Lendaris, "Guest editorial: special issue on adaptive dynamic programming and reinforcement learning in feedback control," *IEEE Transactions on Systems, Man, and Cybernetics B: Cybernetics*, vol. 38, no. 4, pp. 896–897, 2008.

[15] F. L. Lewis and D. Vrabie, "Reinforcement learning and adaptive dynamic programming for feedback control," *IEEE Circuits and Systems Magazine*, vol. 9, no. 3, pp. 32–50, 2009.

[16] X. Xu, L. Zuo, and Z. Huang, "Reinforcement learning algorithms with function approximation: recent advances and applications," *Information Sciences*, vol. 261, pp. 1–31, 2014.

[17] K. G. Vamvoudakis and F. L. Lewis, "Online actor-critic algorithm to solve the continuous-time infinite horizon optimal control problem," *Automatica*, vol. 46, no. 5, pp. 878–888, 2010.

[18] K. G. Vamvoudakis and F. L. Lewis, "Multi-player non-zero-sum games: online adaptive learning solution of coupled Hamilton-Jacobi equations," *Automatica*, vol. 47, no. 8, pp. 1556–1569, 2011.

[19] D. Vrabie and F. Lewis, "Neural network approach to continuous-time direct adaptive optimal control for partially unknown nonlinear systems," *Neural Networks*, vol. 22, no. 3, pp. 237–246, 2009.

[20] L. C. Baird III, "Reinforcement learning in continuous time: advantage updating," in *Proceedings of the IEEE International Conference on Neural Networks (ICNN '94)*, pp. 2448–2453, June 1994.

[21] T. Hanselmann, L. Noakes, and A. Zaknich, "Continuous-time adaptive critics," *IEEE Transactions on Neural Networks*, vol. 18, no. 3, pp. 631–647, 2007.

[22] A. Y. Ng, H. J. Kim, M. I. Jordan, and S. Sastry, "Autonomous helicopter flight via reinforcement learning," *Advances in Neural Information Processing Systems*, vol. 16, 2004.

[23] M. Carreras, J. Yuh, J. Batlle, and P. Ridao, "A behavior-based scheme using reinforcement learning for autonomous underwater vehicles," *IEEE Journal of Oceanic Engineering*, vol. 30, no. 2, pp. 416–427, 2005.

[24] M. A. Kareem Jaradat, M. Al-Rousan, and L. Quadan, "Reinforcement based mobile robot navigation in dynamic environment," *Robotics and Computer-Integrated Manufacturing*, vol. 27, no. 1, pp. 135–149, 2011.

[25] P. G. Balaji, X. German, and D. Srinivasan, "Urban traffic signal control using reinforcement learning agents," *IET Intelligent Transport Systems*, vol. 4, no. 3, pp. 177–188, 2010.

[26] D. Ernst, M. Glavic, and L. Wehenkel, "Power systems stability control: reinforcement learning framework," *IEEE Transactions on Power Systems*, vol. 19, no. 1, pp. 427–435, 2004.

[27] S. Mohagheghi, G. K. Venayagamoorthy, and R. G. Harley, "Adaptive critic design based neuro-fuzzy controller for a static compensator in a multimachine power system," *IEEE Transactions on Power Systems*, vol. 21, no. 4, pp. 1744–1754, 2006.

[28] K. M. Iftekharuddin, "Transformation invariant on-line target recognition," *IEEE Transactions on Neural Networks*, vol. 22, no. 6, pp. 906–918, 2011.

[29] J. del R. Millán, "Reinforcement learning of goal-directed obstacle-avoiding reaction strategies in an autonomous mobile robot," *Robotics and Autonomous Systems*, vol. 15, no. 4, pp. 275–299, 1995.

[30] X. Zhang, H. Zhang, Q. Sun, and Y. Luo, "Adaptive dynamic programming-based optimal control of unknown nonaffine nonlinear discrete-time systems with proof of convergence," *Neurocomputing*, vol. 91, pp. 48–55, 2012.

[31] D. Wang, D. Liu, and Q. Wei, "Finite-horizon neuro-optimal tracking control for a class of discrete-time nonlinear systems using adaptive dynamic programming approach," *Neurocomputing*, vol. 78, no. 1, pp. 14–22, 2012.

[32] X. Xu, Z. Hou, C. Lian, and H. He, "Online learning control using adaptive critic designs with sparse kernel machines," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 24, no. 5, pp. 762–775, 2013.

[33] D. Wang, D. Liu, D. Zhao, Y. Huang, and D. Zhang, "A neural-network-based iterative GDHP approach for solving a class of nonlinear optimal control problems with control constraints," *Neural Computing and Applications*, vol. 22, no. 2, pp. 219–227, 2013.

[34] M. Fairbank, E. Alonso, and D. Prokhorov, "Simple and fast calculation of the second-order gradients for globalized dual heuristic dynamic programming in neural networks," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 23, no. 10, pp. 1671–1676, 2012.

[35] D. Wang, D. Liu, Q. Wei, D. Zhao, and N. Jin, "Optimal control of unknown nonaffine nonlinear discrete-time systems based on adaptive dynamic programming," *Automatica*, vol. 48, no. 8, pp. 1825–1832, 2012.

[36] G. K. Venayagamoorthy, D. C. Wunsch, and R. G. Harley, "Adaptive critic based neurocontroller for turbogenerators with global dual heuristic programming," in *Proceeding of the IEEE Power Engineering Society Winter Meeting*, vol. 1, pp. 291–294, Singapore, January 2000.

[37] J. Peters and S. Schaal, "Natural actor-critic," *Neurocomputing*, vol. 71, no. 7–9, pp. 1180–1190, 2008.

[38] H. Zhang, L. Cui, X. Zhang, and Y. Luo, "Data-driven robust approximate optimal tracking control for unknown general nonlinear systems using adaptive dynamic programming method," *IEEE Transactions on Neural Networks*, vol. 22, no. 12, pp. 2226–2236, 2011.

[39] Z. Hendzel, A. Burghardt, and M. Szuster, "Reinforcement learning in discrete neural control of the underactuated system," *Lecture Notes in Artificial Intelligence*, vol. 7894, pp. 64–75, 2013.

[40] P. Gierlak, M. Szuster, and W. Zylski, "Discrete dual-heuristic programming in 3DOF manipulator control," in *Artifical Intelligence and Soft Computing*, vol. 6114 of *Lecture Notes in Artificial Intelligence*, pp. 256–263, 2010.

[41] Z. Hendzel, "An adaptive critic neural network for motion control of a wheeled mobile robot," *Nonlinear Dynamics*, vol. 50, no. 4, pp. 849–855, 2007.

[42] Z. Hendzel and M. Szuster, "Discrete action dependent heuristic dynamic programming in wheeled mobile robot control," *Solid State Phenomena*, vol. 164, pp. 419–424, 2010.

[43] Z. Hendzel and M. Szuster, "Discrete model-based adaptive critic designs in wheeled mobile robot control," *Lecture Notes in Computer Science*, vol. 6114, no. 2, pp. 264–271, 2010.

[44] Z. Hendzel and M. Szuster, "Discrete neural dynamic programming in wheeled mobile robot control," *Communications in Nonlinear Science and Numerical Simulation*, vol. 16, no. 5, pp. 2355–2362, 2011.

[45] Z. Hendzel and M. Szuster, "Neural dynamic programming in reactive navigation of wheeled mobile robot," in *Artificial Intelligence and Soft Computing*, vol. 7268 of *Lecture Notes in Computer Science*, pp. 450–457, 2012.

[46] J. Giergiel and W. Zylski, "Description of motion of a mobile robot by Maggies Equations," *Journal of Theoretical and Applied Mechanics*, vol. 43, no. 3, pp. 511–521, 2005.

[47] F. L. Lewis, J. Campos, and R. Selmic, *Neuro-Fuzzy Control of Industrial Systems with Actuator Nonlinearities*, Society for Industrial and Applied Mathematics, Philadelphia, Pa, USA, 2002.