Int. J. of Applied Mechanics and Engineering, 2016, vol.21, No.4, pp.867-878
DOI: 10.1515/ijame-2016-0051

APPLICATION OF DIFFERENTIAL GAMES IN MECHATRONIC CONTROL SYSTEM

Z. HENDZEL and P. PENAR*
Faculty of Mechanical Engineering and Aeronautics
Rzeszów University of Technology
ul. Powstańców Warszawy 8, 35-959 Rzeszów, POLAND
E-mail: ppenar@prz.edu.pl

Differential games are a combination of game theory and optimum control methods. Their solutions are based on Bellman's principle of optimality. In this paper, the zero-sum differential game theory has been used for the purposes of controlling a mechatronic object: a single-link manipulator. In this case, analytical solutions are unavailable, thus approximate solutions were used. Two approximation methods were compared with the use of numerical simulations and selected quality indicators. The results confirm previous assumptions and the connection between the differential game theory and H_∞ control problems.

Key words: differential games, optimal control, approximate dynamic programming.

1. Introduction

The differential game theory, combining game theory with optimum control theory, is a generalisation of the min-max optimisation problem. Solutions of differential games are based on Bellman's principle of optimality [1, 2].

Differential games are a subset of dynamic games, in which the controlled object is described by differential equations. In the case of zero-sum differential games, one of the players minimises, and the other maximises the defined value function. The solution of a problem defined this way is a Nash saddle point [3], which is the solution of H_∞ control problems. This fact links the differential game theory with the dissipative system theory [4, 5]. Analytical solutions of zero-sum differential games are only possible if the controlled object is linear, which was utilised, for example, in study [6]. In the nonlinear case, the solution of a zero-sum differential game is approximated using approximate dynamic programming methods [2, 7-9].

The paper presents a single-link manipulator control algorithm [10]. An approximate solution was calculated with the selected zero-sum differential game solution approximation methods (i.e. [7, 11]) taken into account. The solutions obtained were compared using selected quality indicators.

2. H$_\infty$ optimum control of a nonlinear object. Zero-sum differential games

The controlled object is defined [3, 5, 7]

$$\dot{\boldsymbol{x}} = f(\boldsymbol{x}) + g(\boldsymbol{x})\boldsymbol{u} + k(\boldsymbol{x})\boldsymbol{d}\,, \tag{2.1}$$

with an output

[*] To whom correspondence should be addressed

$$\boldsymbol{z} = \boldsymbol{C}\boldsymbol{x} \tag{2.2}$$

where $f(0) = 0$, $\boldsymbol{x}$ is a state dynamic object, $\boldsymbol{C}$ is the output matrix, and f, g, k are non-linear functions. A value function in the following form was correlated with the controlled object (2.1)

$$V(\boldsymbol{x}) = \int_{t_0}^{\infty} \boldsymbol{s}_V(\boldsymbol{x}) = \int_{t_0}^{\infty} [\boldsymbol{x}^T(t)\boldsymbol{Q}\boldsymbol{x}(t) + \boldsymbol{u}^T(t)\boldsymbol{R}\boldsymbol{u}(t) - \gamma \boldsymbol{d}^T(t)\boldsymbol{d}(t)]dt \tag{2.3}$$

where t_0 is the initial time, γ is the gain related to H_∞ control, while $\boldsymbol{Q} = \boldsymbol{C}^T\boldsymbol{C}$ and $\boldsymbol{R}$ are design matrices. The form of the value function (2.3) results from the dissipative system theory, as demonstrated in [4, 5]. Signals $\boldsymbol{u}$ and $\boldsymbol{d}$ are the control signal, which is the minimising player, and the maximising player signal, acting as disturbance, respectively. Output (2.2), designated $\boldsymbol{z}$, determines the value function's value.

For the controlled object (2.1), L_2-gain can be specified:

Theorem 1 [5]**:** In an infinite time horizon, the controlled object (2.1) has gain L_2 lower or equal to γ if

$$\frac{\int_{t_0}^{\infty} [\boldsymbol{x}^T(t)\boldsymbol{Q}\boldsymbol{x}(t) + \boldsymbol{u}^T(t)\boldsymbol{R}\boldsymbol{u}(t)]dt}{\int_{t_0}^{\infty} \boldsymbol{d}^T(t)\boldsymbol{d}(t)dt} \leq \gamma^2, \tag{2.4}$$

H_∞ control involves determining the lowest $\gamma^* > 0$ value such that for any γ, the following inequality is true

$$\gamma > \gamma^* \tag{2.5}$$

where γ^* is the gain determined for signals $\boldsymbol{u}^*$ and $\boldsymbol{d}^*$, which are optimal relative to the defined quality indicator in the form (2.3).

Based on [1, 5], determining gain L_2 for the controlled object (2.3) is equivalent to calculation of the optimum function V. Consequently, a H_∞ control problem is equivalent to minimisation of the value function V. Such an approach reduces a L_2 stability problem to the zero-sum differential game theory.

2.1. Zero-sum differential game

As shown in paper [3], differential game problems are defined as follows: the solution of a zero-sum differential game where the controlled object is defined by Eq.(2.1), the output Eq.(2.2), and the value function takes the form (2.3), are such signals $\boldsymbol{u}^*$ and $\boldsymbol{d}^*$, for which the following inequality is true

$$V\left(\boldsymbol{x}, \boldsymbol{u}^*, \boldsymbol{d}\right) \leq V\left(\boldsymbol{x}, \boldsymbol{u}^*, \boldsymbol{d}^*\right) \leq V\left(\boldsymbol{x}, \boldsymbol{u}, \boldsymbol{d}^*\right). \tag{2.6}$$

In other words, the solution of a zero-sum differential game is a pair of signals $\left(\boldsymbol{u}^*, \boldsymbol{d}^*\right)$ which define a saddle point, referred to as a Nash saddle point [3]. A saddle point solution is the optimum game strategy for both players.

Zero-sum differential game problems can be expressed using the Hamilton-Jacobi-Isaacs (HJI) equation, i.e.,

$$0=\min_{\boldsymbol{u}(t)}\max_{\boldsymbol{d}(t)}\left\{\boldsymbol{s}_V(\boldsymbol{x},\boldsymbol{u},\boldsymbol{d})+\frac{dV}{d\boldsymbol{x}}\left[f(\boldsymbol{x})+g(\boldsymbol{x})\boldsymbol{u}+k(\boldsymbol{x})\boldsymbol{d}\right]\right\}, \tag{2.7}$$

or

$$0=\min_{\boldsymbol{d}(t)}\max_{\boldsymbol{u}(t)}\left\{s_V(\boldsymbol{x},\boldsymbol{u},\boldsymbol{d})+\frac{dV}{d\boldsymbol{x}}\left[f(\boldsymbol{x})+g(\boldsymbol{x})\boldsymbol{u}+k(\boldsymbol{x})\boldsymbol{d}\right]\right\}. \tag{2.8}$$

The HJI equation defines a sufficient condition for an optimum. As a consequence of Eqs (2.7) and (2.8), the saddle point $\left(\boldsymbol{u}^*,\boldsymbol{d}^*\right)$ is defined as

$$\boldsymbol{u}^*(t)=-\frac{1}{2}\boldsymbol{R}^{-1}g(\boldsymbol{x})^T\frac{dV^*(\boldsymbol{x})}{d\boldsymbol{x}}, \tag{2.9}$$

and

$$\boldsymbol{d}^*(t)=\frac{1}{2\gamma^2}k(\boldsymbol{x})^T\frac{dV^*(\boldsymbol{x})}{d\boldsymbol{x}} \tag{2.10}$$

where $V^*(\boldsymbol{x})$ designates the optimum function value.

It has been widely indicated [1, 2, 7] that solving zero-sum differential games using Eqs (2.9) and (2.10) is very difficult due to the value function gradient being unknown. Analytical solutions are possible in linear cases. Zero-sum differential game solutions are then reduced to the Riccati solution, as demonstrated for a discrete problem in paper [6].

In non-linear cases, a frequently utilised solution is to approximate the value function using adaptive structures, such as neural networks [2, 7–9].

3. Approximate solutions

3.1. Actor-critic structure

As shown in paper [7], zero-sum differential games can be solved using sequential policy iteration (PI) algorithms. Their drawback is the need to execute internal loops in which the maximising player's signal is determined, while maintaining a constant control signal (the minimising player). Only when the internal loop is complete is the next control signal value determined. The necessity to operate online enforces the use of other approximation methods, e.g., the widely utilised author-critic structures [11].

As discussed in paper [7], value function assessment utilising the critic's neural network (NN) has the following form

$$\hat{V}=\hat{\boldsymbol{W}}_1^T\psi(\boldsymbol{x}) \tag{3.1}$$

where $\hat{\boldsymbol{W}}_1$ is the NN weight approximate, while $\boldsymbol{\psi}(\boldsymbol{x})=[\psi_1(\boldsymbol{x}),\psi_2(\boldsymbol{x}),...,\psi_N(\boldsymbol{x})]^T$ is the basic function vector. The value function reaches the lowest value when the Hamiltonian

$$H(\boldsymbol{x},\hat{\boldsymbol{W}}_1,\boldsymbol{u},\boldsymbol{d})=\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x}+\boldsymbol{u}^T\boldsymbol{R}\boldsymbol{u}-\gamma\boldsymbol{d}^T\boldsymbol{d}+\left(\hat{\boldsymbol{W}}_1^T\frac{d\boldsymbol{\psi}(\boldsymbol{x})}{d\boldsymbol{x}}\right)\left(f(\boldsymbol{x})+g(\boldsymbol{x})\boldsymbol{u}+k(\boldsymbol{x})\boldsymbol{d}\right), \tag{3.2}$$

calculated from Eq.(2.7), equals zero. The task of the NN approximating value function is to minimise the error defined as

$$E=\frac{1}{2}H^{2}. \tag{3.3}$$

Equation (3.3) forms the basis for determining the weight adaptation rule, which takes the form of a modified Levenberg-Marquardt method [7], i.e.,

$$\dot{\hat{\boldsymbol{W}}}_{1}=-\eta\frac{\sigma_{1}}{\left(1+\sigma_{1}^{T}\sigma_{1}\right)}\left[\sigma_{1}^{T}\hat{\boldsymbol{W}}_{1}+\boldsymbol{x}^{T}\boldsymbol{Q}\boldsymbol{x}+\boldsymbol{u}^{T}\boldsymbol{R}\boldsymbol{u}-\gamma\boldsymbol{d}^{T}\boldsymbol{d}\right] \tag{3.4}$$

where $\sigma_{1}=\nabla\psi(\boldsymbol{x})\left[f(\boldsymbol{x})+g(\boldsymbol{x})\boldsymbol{u}+k(\boldsymbol{x})\boldsymbol{d}\right]$, $\nabla\psi(\boldsymbol{x})=\dfrac{d\psi(\boldsymbol{x})}{d\boldsymbol{x}}$ and η is the learning coefficient.

Using value function gradient approximations, relations (2.9) and (2.10) are expressed as

$$\hat{\boldsymbol{u}}(t)=-\frac{1}{2}\boldsymbol{R}^{-1}g(\boldsymbol{x})^{T}\nabla\psi(\boldsymbol{x})\hat{\boldsymbol{W}}_{1}, \tag{3.5}$$

and

$$\hat{\boldsymbol{d}}(t)=\frac{1}{2\gamma^{2}}k(\boldsymbol{x})^{T}\nabla\psi(\boldsymbol{x})\hat{\boldsymbol{W}}_{1}. \tag{3.6}$$

Given Eqs (3.5) and (3.6), the weight adaptation rule (3.4) of the critic's NN takes the following form

$$\dot{\hat{\boldsymbol{W}}}_{1}=-\eta\frac{\hat{\sigma}_{1}}{\left(1+\hat{\sigma}_{1}^{T}\hat{\sigma}_{1}\right)}\left[\sigma_{1}^{T}\hat{\boldsymbol{W}}+\boldsymbol{x}^{T}\boldsymbol{Q}\boldsymbol{x}+\hat{\boldsymbol{u}}^{T}\boldsymbol{R}\hat{\boldsymbol{u}}-\gamma\hat{\boldsymbol{d}}^{T}\hat{\boldsymbol{d}}\right] \tag{3.7}$$

where $\hat{\sigma}_{1}=\nabla\psi(\boldsymbol{x})\left[f(\boldsymbol{x})+g(\boldsymbol{x})\hat{\boldsymbol{u}}+k(\boldsymbol{x})\hat{\boldsymbol{d}}\right]$.

3.2. Recurring least squares (RLS) method

Another approach to zero-sum differential games is the use of the iterative least squares method in the problem identification area [12]. The Hamiltonian's form (3.2) indicates that it is a linear function of the critic's weights. Therefore, application of the least squares method is viable.

The value function is defined as

$$\hat{V}=\hat{\boldsymbol{W}}_{2}^{T}\psi(\boldsymbol{x}). \tag{3.8}$$

In accordance with study [12], the critic's NN weight adaptation process was executed using the iterative normalised least squares method, where the weight adaptation rule has the following form

$$\dot{\hat{\boldsymbol{W}}}_{2}^{T}=\xi\boldsymbol{G}\frac{\sigma_{1}}{1+\nu\sigma_{1}^{T}\boldsymbol{G}\sigma_{1}}H\left(\boldsymbol{x},\hat{\boldsymbol{W}}_{2},\boldsymbol{u},\boldsymbol{d}\right). \tag{3.9}$$

The $\xi,\nu>0$ indicators present in Eq.(3.9) are design parameters, while $\boldsymbol{G}$ is a symmetrical matrix calculated from the following equation

$$\dot{\boldsymbol{G}} = \xi \boldsymbol{G} \frac{\sigma_1 \sigma_1^T}{1 + \nu \sigma_1^T \boldsymbol{G} \sigma_1} \boldsymbol{G} , \tag{3.10}$$

while $\boldsymbol{G}(0) = \eta_G \boldsymbol{I}, \qquad \eta_G >> 0 .$

4. Single-link manipulator control

The single-link manipulator, whose scheme is presented in Fig.1, remains in rotating motion around the z axis which passes through point A. Figure 1 indicates the manipulator motion equation, which has the following form

$$J_A \ddot{\alpha} = u(t) - \frac{1}{2} G l \cos(\alpha) - \kappa \dot{\alpha} - d(t) \tag{4.1}$$

where α is the robot's rotation angle, $\kappa \dot{\alpha}$ is the movement resistance term (κ is constant), u is the driving moment which constitutes control, d is the disturbance moment, $G = m g_m$ is the weight of the arm with m mass (g_m - gravitational constant), l is the arm length, and J_A is the arm's moment of interia relative to the rotation axis passing through point A.

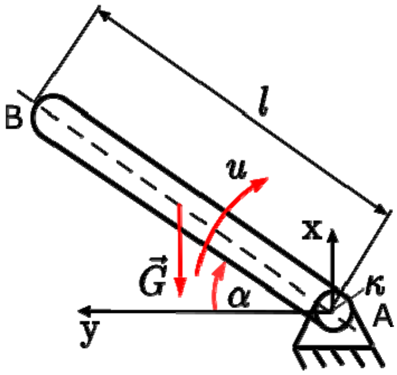

Fig.1. Single-link manipulator scheme.

Based on the parametric identification process [13], Eq.(4.1) is given the following form

$$a_1 \ddot{\alpha} - a_2 \dot{\alpha} + a_3 g \cos(\alpha) = u - d \tag{4.2}$$

where the obtained values of the a_i coefficients are provided in Tab.1

Tab.1. Parameter values a_i .

a_1	0.1523
a_2	0.4648
a_3	0.0063

In the following part of the study, differential game theory was utilised to control tracking movement of the manipulator. To this end, control error was defined

$$e = \alpha - \alpha_d\,, \tag{4.3}$$

and

$$\dot{e} = \dot{\alpha} - \dot{\alpha}_d \tag{4.4}$$

where α_d , $\dot{\alpha}_d$ are defined kinematic parameters of the arm's movement. A generalised error

$$s = \dot{e} + \lambda e \tag{4.5}$$

where λ is the design coefficient, was assumed. After differentiating the Eq.(4.5) and using Eqs (4.2) - (4.4), the following was obtained

$$\dot{s} = -\frac{a_2}{a_1}s + N\left(e, \dot{e}, \alpha_d, \dot{\alpha}_d\right) + \frac{1}{a_1}u - \frac{1}{a_1}d \tag{4.6}$$

where d is the signal generated by the maximising player, who serves as disturbance, and $N(e,\dot{e},\alpha_d\dot{\alpha}_d)$ is a function defined as

$$N(e,\dot{e},\alpha_d\dot{\alpha}_d) = --\frac{a_2}{a_1}(\alpha_d - \lambda e) - \frac{a_3}{a_1}g\cos(e + \alpha_d) - \ddot{\alpha}_d + \lambda\dot{e}\,. \tag{4.7}$$

Equation (4.7) is a description of the robot arm's movement dynamics in a generalised error space. Assuming a_1, a_2 and a3, parameter values as given in Tab.1 and assuming the defined trajectory of the selected point B , as well as its velocity $\bar{v}_B$ the tracking control problem using a zero-sum differential game was defined. When the inverse kinematics problem was solved, angular coordinates of the defined motion trajectory were obtained, and the trajectory's kinematic parameters could be defined as

$$\begin{aligned} \alpha_d &= A\cos(\omega t), \\ \dot{\alpha}_d &= -A\omega\sin(\omega t), \\ \ddot{\alpha}_d &= -A\omega^2\cos(\omega t) \end{aligned} \tag{4.8}$$

where A and ω are selected constant.

4.1. Actor-critic structure simulations

Considering the form of the value function (2.3), selecting $A = \pi/10$, $\omega = 3$, $\gamma = 0.51$, $R=0.2$, $Q=0.2$, $\eta = 100$, $\lambda = 1$, $\alpha(0) = \pi / 10$, $\dot{\alpha}(0) = 0$, and considering the dynamics description, i.e., Eq.(4.6), a numerical simulation of tracking control was performed using an author-critic structure related to the zero-sum differential game approximation. The simulation was performed using the Matlab/Simulink suite using Euler's integration method with a $h = 0.001$ step. Selecting the vector of the basic functions describing NN neurons

$$\boldsymbol{\psi}(\boldsymbol{x}) = \left[s^2, e^2, \dot{e}^2\right]^T, \tag{4.9}$$

whose task is to approximate the value function, and assuming $\hat{\boldsymbol{W}}_1(0) = \left[0.18, 4.8, 1.2\right]^T$, a numerical simulation of a differential game and H_∞ control was conducted

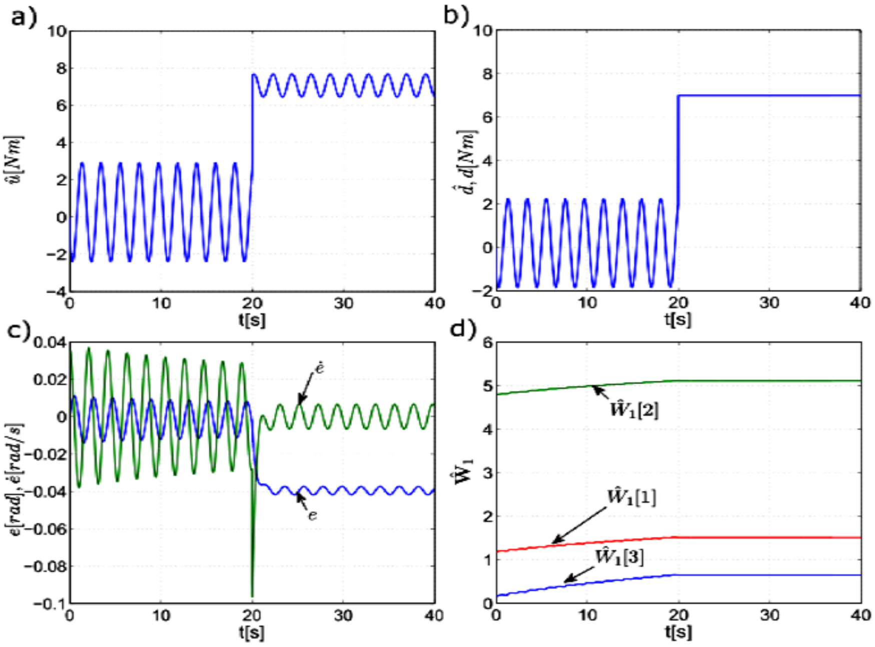

Fig.2. The actor-critic algotithm result: (a) behaviour of $\hat{u}$ which is the minimising player; (b) behaviour of maximising player signal $\hat{d}$ and disturbance step d; (c) behaviour of tracking control error e and $\dot{e}$; (d) behaviour of weight $\hat{W}_1$.

Figure 2a presents the control behaviour for worst case disturbance (Fig.2b) of the Nash saddle point solution. This process is executed for $t \in \langle 0, 20 \rangle$. The obtained representation error for the defined trajectory during the execution of the differential game is presented in Fig.2c. Execution of the differential game procedure requires approximating the value function using NN in an actor-critic structure by an appropriate NN weight adaptation process. The obtained weights are presented in Fig.2d.

The discussed zero-sum differential game results were used to solve an H_∞ control problem. In order to verify this control, step disturbance with $d = 7[Nm]$ amplitude was introduced in the second part of the simulation (for $t \geq 20$). They can be interpreted as the manipulator moving additional weight.

While the disturbance was present, increased control signal amplitude was noted, as shown in Fig.2a. The error and its derivative, obtained during that time, are presented in Fig.2c.

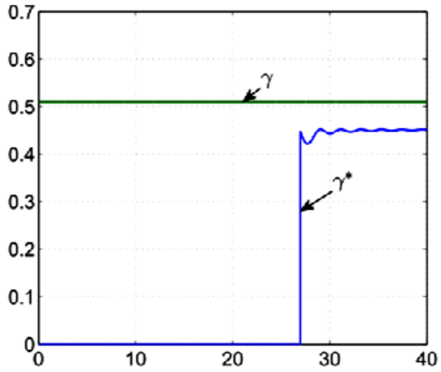

Fig.3. Behaviours of the gain γ and γ^*.

As shown in Fig.3, conditions (4.5) is met, meaning that the designed system is not susceptible to interference occurring during the execution of the control process. This means that the designed system is stable in the sense L_2 [5]. It is an input-output stability, which is a generalisation of the linear system stability method.

Since the value function can be approximated using a normalised RLS algorithm, the subsequent part of this paper presents the zero-sum differential game and H_∞ control results obtained using this approach.

4.2. RLS algorithm simulations

In this section, the critic's neural network weight learning algorithm was executed using the procedure described in section 3.2. With the dynamic equation of the manipulator arm's movement in the generalised tracking error space (Eq.(4.6)) known, the RLS method was used to obtain the approximate solution of a zero-sum differential game. Keeping the parameter values from section 4.1 and selecting $\boldsymbol{G}(0)=22\boldsymbol{I}_{3x3}$, $\eta=0.35,$, $\nu=0.01$, $\hat{\boldsymbol{W}}_2(0)=\left[4,0.08,1.6\right]^T$ and using the weight adaptation rule $\hat{\boldsymbol{W}}_2$ defined in Eq.(4.4), a numerical simulation of a differential game and H_∞ control was performed.

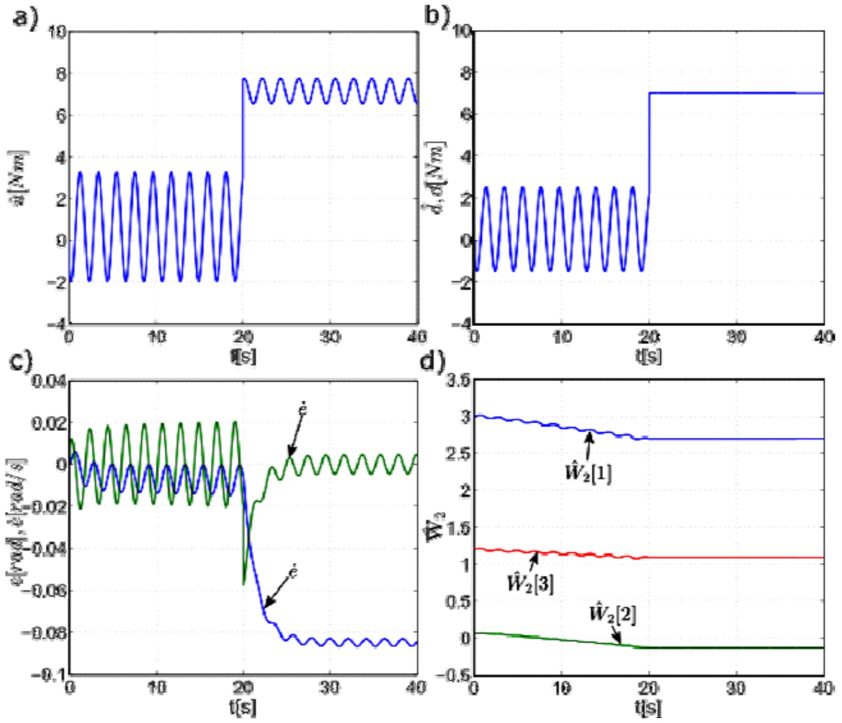

Fig.4. The normalised RLS algotithm result: (a) behaviour of $\hat{u}$ which is the minimising player; (b) behaviour of maximising player signal $\hat{d}$ and disturbance step d; (c) behaviour of tracking control error e and $\dot{e}$; (d) behaviour of weight $\hat{\boldsymbol{W}}_2$.

Similarly to the previous simulation, a differential game was simulated using a normalised RLS algorithm during the first phase, i.e., for $t\in\langle 0,20\rangle$. Figure 4a presents the control signal $\hat{u}$, which is a response to the worst case disturbance shown in Fig.4b. During the differential game, tracking errors e and $\dot{e}$ are limited, as shown in Fig.4c. Additionally, NN weights (Fig.4d) are adapted using Eq.(3.9) in this simulation phase.

In the second simulation phase, i.e., during time $t \in \langle 20, 40 \rangle$, the determined NN weights $\hat{\boldsymbol{W}}_2$, which approximate the value function gradient, were used to solve the H_∞ problem. Similarly to section 4.1, step disturbance $d = 7[Nm]$ was added to the system. The control response $\hat{u}$ to the additional interference is shown in Fig.4a. Tracking errors e and $\dot{e}$ during the presence of the discrete interference are shown in Fig.4c.

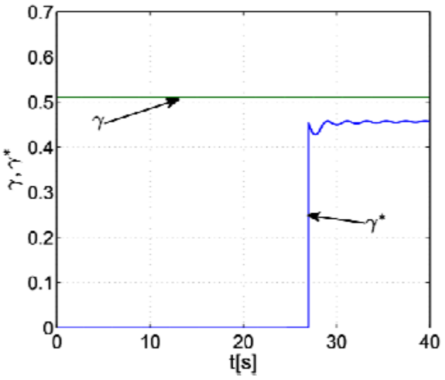

Fig.5. Behaviours of the gain γ and γ^* for normalised RLS algorithm.

The theory indicates that H_∞ control is stable when $\gamma > \gamma^*$. Figure 5 indicates that $\gamma > \gamma^*$, which means that the designed control system is stable in the sense L_2.

5. Comparison of results

In order to compare the zero-sum differential game approximate solution methods presented in this paper, a number of quality indicators were introduced. Given that a numerical integration procedure was utilised in the simulation, the presented methods can be assessed using similar quality indicators as in paper [14]:

- module of the maximum rotation angle $|e_{\max}|$ and angular velocity $|\dot{e}_{\max}|$ tracking error,
- mean square error root of the rotation angle $\varepsilon = \sqrt{\frac{1}{n}\sum_{k=1}^{n} e_k^2}$ and angular velocity $\dot{\varepsilon} = \sqrt{\frac{1}{n}\sum_{k=1}^{n} \dot{e}_k^2}$ tracking error,
- maximum value of the generalised tracking error $|s_{\max}|$,
- mean square error root of the generalised tracking error $\sigma = \sqrt{\frac{1}{n}\sum_{k=1}^{n} s_k^2}$

where n is the sample number

The values of the selected quality indicators for section 4.1 (Levenberg-Marquardt algorithm) and 4.2 (RLS algorithm) simulations across time $t \in \langle 20, 40 \rangle$ are summarised in Tab.2 and visualised in the bar chart (Fig.6a).

Table 2. Values of the selected quality indicators.

	$e_{\max}$	$\dot{e}_{\max}$	ε	$\dot{\varepsilon}$	$s_{\max}$	σ
LM algorithm	*0.0038*	*0.0068*	*0.0371*	*0.0107*	*0.0303*	*0.0394*
RLS algorithm	*0.0060*	*0.0064*	*0.0803*	*0.0109*	*0.0257*	*0.0833*

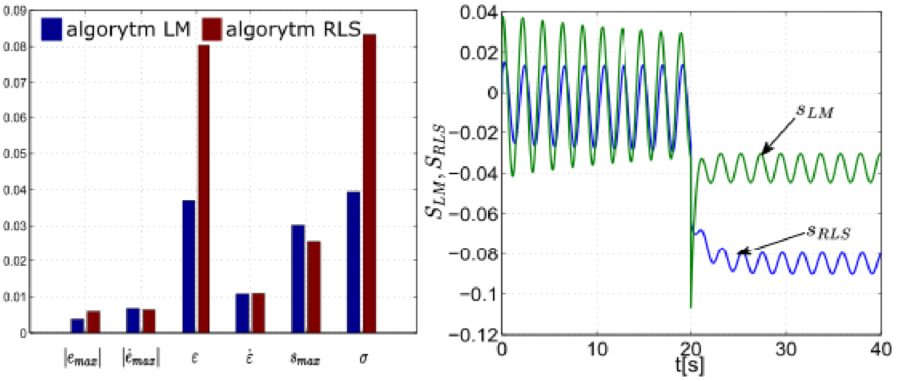

Fig.6. (a) Comparing selected quality indicators and (b) comparing generalised errors s for methods presented in sections 4.1 and 4.2.

Considering the selected quality indicators, it can be concluded that the control quality measured using the selected quality indicators is higher when using the procedure described in section 4.2. This observation is confirmed by Fig.6b, where generalised errors s for methods presented in sections 4.1 (s_{LM}) and 4.2 (s_{RLS}) are summarised.

6. Summary

The paper presents an example of differential game theory application in non-linear object control, which is equivalent to solving an H_∞ control problem. Furthermore, it is known that meeting the $\gamma > \gamma^*$ condition means that the designed system is stable in the sense L_2 [5]. It is an input-output stability, which is a generalisation of the linear system stability method.

The paper compares two methods of approximating the HJI equation. In both cases, H_∞ control verification confirmed that the $\gamma > \gamma^*$ condition was met. Additionally, considering the quality indicators calculated in section 5, it was concluded that the method presented in section 4.1 is slightly better in minimising the tracking error.

Computer simulations led to obtaining a stable solution in spite of varying operating conditions of the analysed single-link manipulator. These results confirm the theoretical assumptions concerning the HJI equation solution stability [7].

Nomenclature

A, ω – selected constant
a_i – constant parameter values
$\boldsymbol{C}$ – output matrix
$\boldsymbol{d}$ – maximising player
$\boldsymbol{d}^*$ – optimal maximising player

$\hat{\boldsymbol{d}}$	– approximation maximising player
e	– control error
$\lvert e_{\max} \rvert$	– module of the maximum rotation angle
f,g,k	– nonlinear functions
$G = mg_m$	– weight of the arm with m mass (g_m - gravitational constant)
h	– integration step
J_A	– arm's moment of interia
l	– arm length
s	– generalized control error
$\lvert s_{\max} \rvert$	– maximum value of the generalised tracking error
$\boldsymbol{u}$	– minimising player
$\boldsymbol{u}^*$	– optimal minimising player
$\hat{\boldsymbol{u}}$	– approximation minimising player
$\boldsymbol{Q},\boldsymbol{R}$	– design matrices
V	– value function
$\hat{V}$	– approximation value function
$\hat{\boldsymbol{W}}_i$	– NN weight approximate
$\boldsymbol{x}$	– state dynamic object
α , α_d	– robot's rotation angle
γ	– gain related to H_∞ control
ε	– mean square error root of the rotation angle
η	– learning coefficient
$\kappa\dot{\alpha}$	– movement resistance term (κ is constant)
λ	– design coefficient
ξ,ν , η_G	– design parameters
σ	– mean square error root of the generalised tracking error
$\boldsymbol{\psi}(\boldsymbol{x}) = [\psi_1(\boldsymbol{x}),\psi_2(\boldsymbol{x}), ,\dots,\psi_N(\boldsymbol{x})]^T$	– basic function vector

References

[1] Abu-Khalaf M., Huang J. and Lewis Frank. L. (2006): *Nonlinear H_2/H_∞ Constrained Feedback Control.* – London: Springer.

[2] Johnson Marcus A. (2011): *Differential game-based control methods for uncertain continuous-time nonlinear systems*. – Phd Thesis, University of Florida.

[3] Starr A.W. and Ho Y.C. (1969): *Nonzero-sum differential games*. – Journal of Optimization Theory and Applications, vol.3, No.3, pp.184–206.

[4] Willems Jan C. (1972): *Dissipative dynamical systems part I: General theory*. – Archive for Rational Mechanics and Analysis, vol.45, No.5, pp.321–351.

[5] Van der Schaft A.J. (1992): *L_2-gain analysis of nonlinear systems and nonlinear state feedback H_∞ control*. – IEEE Trans. on Autom. Control, vol.37, No.6, pp.770-784 1992.

[6] Hendzel Z. and Penar P. (2016): *The used of differential game theory in the control wheeled mobile robot motor* (in Polish). – Przegląd Mechaniczny, vol.75, No.1–2, pp.25–31.

[7] Kyriakos G. Vamvoudakis and Lewis Frank L. (2012): *Online solution of nonlinear two-player zero-sum games using synchronous policy iteration*. – International Journal of Robust and Nonlinear Control, vol.22, pp.1460–1483.

[8] Huai-Ning Wu and Biao Luo (2012): *Neural network based online simultaneous policy update algorithm for solving the HJI equation in nonlinear* H_∞ *control*. – IEEE Transactions Neural Networks and Learning Systems, vol.23, No.12, pp.1884–1895.

[9] Yasini S., Naghibi Sistani M.B. and Karimpour A. (2014): *Policy Iteration Algorithm Based on Experience Replay to Solve* H_∞ *Control Problem of Partially Unknown Nonlinear Systems*. – Int. J. Control. Autom. Syst., European Control Conference.

[10] Hendzel. and Gierlak P. (2011): *Control of Wheeled and Manipulators Robots* (in Polish). – Rzeszow: OWPRz.

[11] Fei-Yue Wang, Zhang H. and Liu D. (2009): *Adaptive dynamic programming: An introduction*. – IEEE Computational Intelligence Magazine, vol.4, No.2, pp.39–47.

[12] Shankar S. and Bodson M. (1989): *Adaptive Control. Stability, Convergence and Robustness*. – New Jersey: Prentice Hall.

[13] Żylski W. and Gierlak P. (2010): *Modelling of movement of selected manipulator* (in Polish). – Acta Mechanica at Automatica, vol.4, No.1, pp.112-119.

[14] Szuster M. (2012): *Generation and Realization of Wheeled Mobile Robot Trajectory Using Neural Dynamic Programming* (in Polish). – PhD Thesis, Rzeszow University of Technology.

Received: August 13, 2016
Revised: September 14, 2016