Доклади на Българската академия на науките
Comptes rendus de l'Académie bulgare des Sciences

Tome 67, No 8, 2014

SCIENCES ET INGENIERIE
Automatique et informatique

Dedicated to the 145th Anniversary of the Bulgarian Academy of Sciences

LINEAR-QUADRATIC CONTROL OF A TWO-WHEELED ROBOT

Tsonio Slavov, Jordan Kralev, Petko Petkov

(*Submitted on June 19, 2014*)

Abstract

In this paper we present the design and experimentation of a control system of two-wheeled robot which implements a Linear-Quadratic Regulator (LQR) for vertical stabilization and a proportional-integral (PI) controller of the robot rotation around the vertical axis. Due to the lack of accurate analytical robot model, the control system design is done by using a model built with the aid of an identification procedure. Two discrete-time Kalman filters of 17th order and of 2nd order are implemented to estimate the plant state in presence of several noises. A software in MATLAB®/Simulink® environment is developed for generation of control code which is embedded in the Texas Instruments Digital Signal Controller TMS320F28335. Results from the simulation of the closed-loop system as well as experimental results obtained during the real implementation of the controller designed are given.

Key words: real-time control, two-wheeled robot, LQR control, embedded system, digital signal processor

1. Introduction. Two-wheeled robots have several applications which make them interesting from theoretical and practical point of view. The most popular commercial product, built on the idea of self balancing two-wheeled robot is the Segway® Personal Transporter (PT), produced by Segway Inc., USA [1]. Some of the Segway® PTs have maximum speed of 20 km/h and can travel as far as 38 km on a single battery charge. The self-balancing two-wheeled robot NXTway-GS [2], built on the basis of the LEGO® Mindstorms NXT developer kit, is widely used in education. Also, the telepresence and video conferencing two-wheeled robot Double® [3] is recently very popular.

The research presented in this paper is done under Project No 142PD0008-08 funded by the Scientific and Research Unit of the Technical University of Sofia.

The two-wheeled robots have dynamics which is similar to the inverted pendulum dynamics so that they are inherently unstable and should be stabilized around the vertical position using a control system. Linear-quadratic or proportional-integral-derivative (PID) control laws are usually implemented in order to achieve vertical stabilization and desired position in the horizontal plane [4].

In this paper we present the design and experimentation of a control system of two-wheeled robot which implements a Linear-Quadratic Regulator for vertical stabilization and a proportional-integral controller of the robot rotation around the vertical axis. Due to the lack of accurate analytical robot model, the control system design is done by using a model built with the aid of an identification procedure. Two discrete-time Kalman filters of 17th order and of 2nd order are implemented to estimate the plant state in presence of several noises. A software in MATLAB®/Simulink® environment is developed for generation of control code which is embedded in the Texas Instruments Digital Signal Controller TMS320F28335. Results from the simulation of the closed-loop system as well as experimental results obtained during the real-time implementation of the controller designed are given. The experimental results confirm the efficiency of the technical solution implemented.

2. Plant description. The general view of the two-wheeled robot in self-balancing mode is shown in Fig. 1*a*.

The robot is equipped with two servo drives for actuation, micro-electromechanical (MEM) inertial sensor ADIS16350 for measuring the angular velocities $\dot{\phi}$ and $\dot{\psi}$ of robot body in the vertical plane and around the vertical axis, respectively, quadrature encoders for measuring the position of the wheels and a digital signal controller Texas Instruments TMS320F28335 implementing a discrete real-time stabilization algorithm with sampling period $T_0 = 0.005$ s. The robot balancing is achieved by rotating the wheels in appropriate direction. The computation of control actions to both DC brushed drive motors is realized in single precision on the basis of signals from the gyro sensor measuring the angular rate (and, after integration, the tilt angle ϕ) and signals from rotary encoders measuring the wheels rotation angles. The control of the DC motors is executed by Pulse Width Modulated (PWM) signals. Since the gyroscope measurements are subject to bias, random walk and noise, the body tilt angle and the wheel angles are estimated by using a 17th order Kalman filter. Additional 2nd order Kalman filter is used to estimate the yaw angle ψ.

3. Controller design. The block-diagram of the two-wheeled robot control system is shown in Fig. 1*b*. With respect to the stabilization in upper equilibrium state and to the control of forward-backward motion, the robot is described by Auto Regressive Moving Average with eXternal input (ARMAX) and Box-Jenkins (BJ) discrete-time models. The ARMAX model of 7th order with structure parameters $na = 7$, $nb = 7$, $nc = 7$, $nk = 3$ describes the link between the control

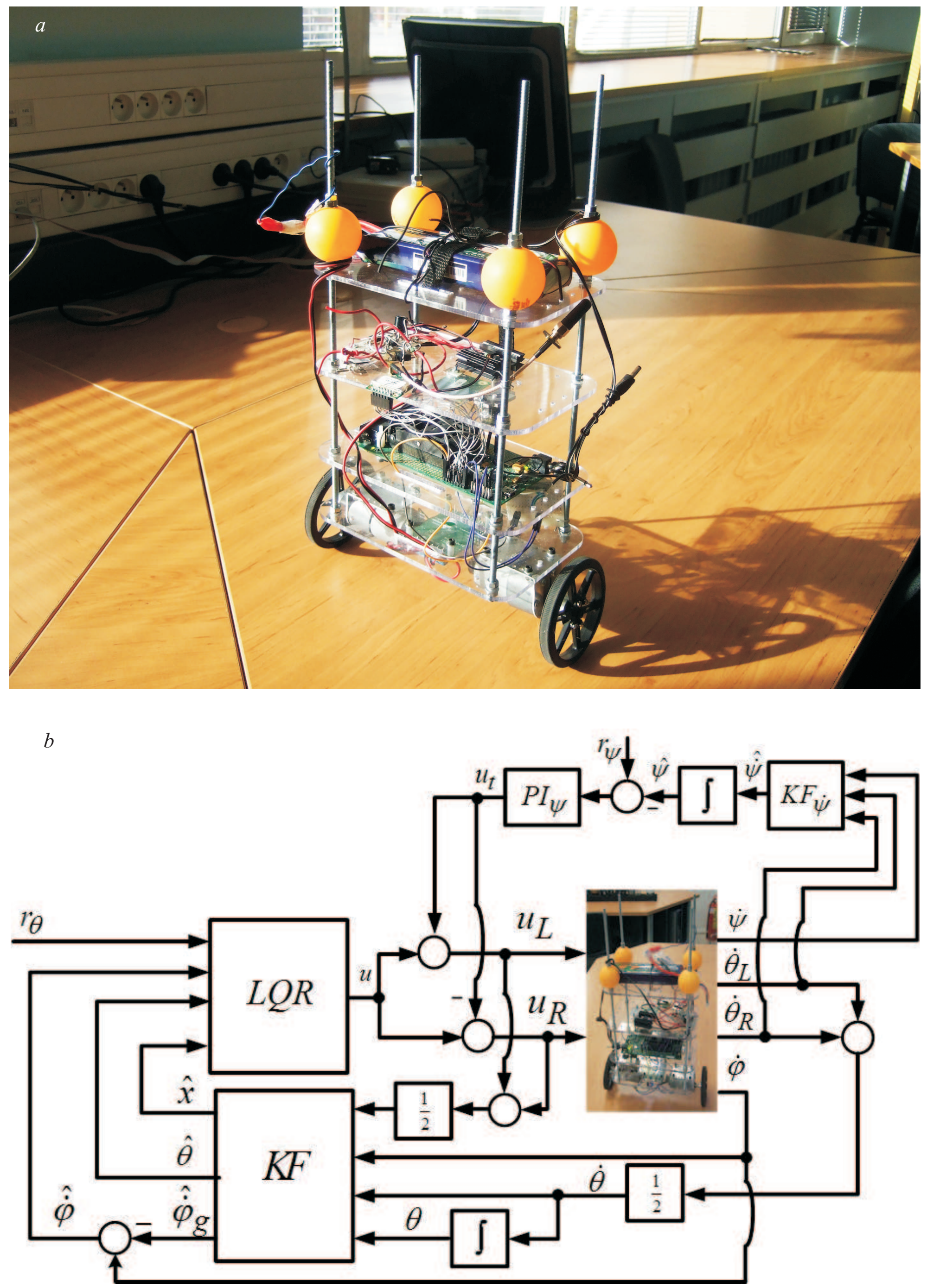


Fig. 1. *a*) Two-wheeled robot in self-balancing mode, *b*) Block-diagram of the control system

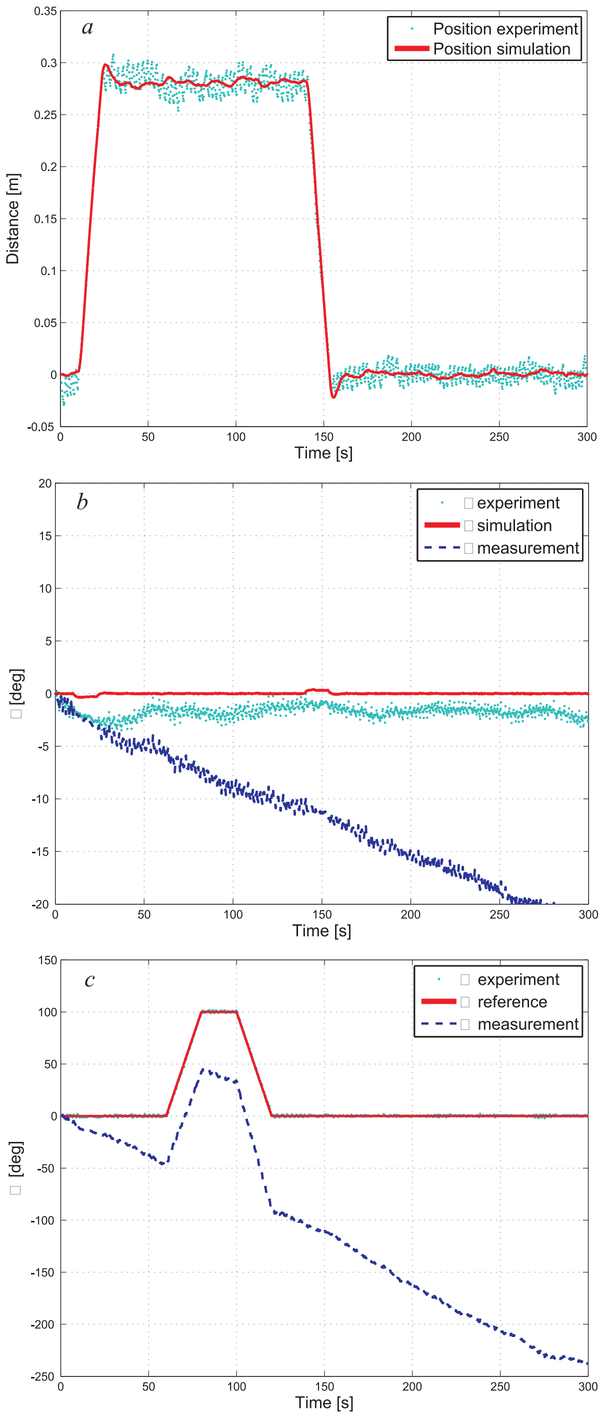


Fig. 3. *a*) Tracking position reference, *b*) Body angle variation, *c*) Tracking yaw angle reference

signal u and the rate $\dot{\phi}$, while the BJ model of 3rd order with structure parameters $nb = 3$, $nf = 3$, $nc = 3$, $nd = 3$, $nk = 1$ gives the link between the rates $\dot{\phi}$ and $\dot{\theta} = (\dot{\theta}_L + \dot{\theta}_R)/2$ where $\dot{\theta}_L$ and $\dot{\theta}_R$ are the angular velocities of left and right wheels, respectively.

The ARMAX and BJ nominal models are represented in the state space as

$$\begin{aligned} x_{\dot{\phi}}(k+1) &= A_{\dot{\phi}} x_{\dot{\phi}}(k) + B_{\dot{\phi}} u(k) + J_{\dot{\phi}} \nu_{\dot{\phi}}(k), \\ \dot{\phi}(k) &= C_{\dot{\phi}} x_{\dot{\phi}}(k) + H_{\dot{\phi}} \nu_{\dot{\phi}}(k), \end{aligned} \tag{1}$$

$$\begin{aligned} x_{\dot{\Theta}}(k+1) &= A_{\dot{\Theta}} x_{\dot{\Theta}}(k) + B_{\dot{\Theta}} u(k) + J_{\dot{\Theta}} \nu_{\dot{\Theta}}(k), \\ \dot{\Theta}(k) &= C_{\dot{\Theta}} x_{\dot{\Theta}}(k) + H_{\dot{\Theta}} \nu_{\dot{\Theta}}(k), \end{aligned} \tag{2}$$

where $x_{\dot{\phi}}$, $x_{\dot{\Theta}}$ are state vectors with dimensions 9 and 6, respectively, $\nu_{\dot{\phi}}$, $\nu_{\dot{\Theta}}$ are discrete-time white gaussian noises with variances $D_{\dot{\phi}} = D_{\dot{\Theta}} = 1$, and $A_{\dot{\phi}}$, $B_{\dot{\phi}}$, $J_{\dot{\phi}}$, $C_{\dot{\phi}}$, $H_{\dot{\phi}}$, $A_{\dot{\Theta}}$, $B_{\dot{\Theta}}$, $J_{\dot{\Theta}}$, $C_{\dot{\Theta}}$, $H_{\dot{\Theta}}$ are constant matrices with corresponding dimensions containing the model parameters. The noises $\nu_{\dot{\phi}}$, $\nu_{\dot{\Theta}}$ are obtained during the identification procedure and reflect the uncertainty in the model found. These noises are represented as models with input multiplicative uncertainty which are used in the analysis in frequency range.

The non-measurable wheels angle is described by

$$\Theta(k+1) = \Theta(k) + T_0 \dot{\Theta}(k) = \Theta(k) + T_0 C_{\dot{\Theta}} x_{\dot{\Theta}}(k) + T_0 H_{\dot{\Theta}} \nu_{\dot{\Theta}}(k). \tag{3}$$

It is also appropriate to include the equations

$$x_{\dot{\phi}_i}(k+1) = x_{\dot{\phi}_i}(k) - T_0 \dot{\phi}(k) \tag{4}$$

$$x_i(k+1) = x_i(k) + T_0(r_\Theta(k) - \Theta(k)), \tag{5}$$

where r_Θ is the wheels reference angle. These equations allow to compute approximations of the discrete-time integrals of $\dot{\phi}$ and tracking error $r_\Theta - \Theta$, respectively. Both integrals are used in the design of linear-quadratic regulator which ensures efficient stabilization in the vertical plane and zero steady-state tracking error.

In this way we obtain the full plant equations of 18th order in the form

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) + J\nu(k), \\ y(k) &= Cx(k) + H\nu(k), \end{aligned} \tag{6}$$

where

$$x = [x_{\dot{\phi}}^T \; x_{\dot{\Theta}}^T \; \Theta \; x_{\dot{\phi}_i} \; x_i]^T, \quad y = [\dot{\phi} \; \dot{\Theta} \; \Theta \; \phi]^T, \quad v = [v_{\dot{\phi}} \; v_{\dot{\Theta}}]^T$$

and the matrices A, B, J, C, H are obtained combining equations (1)–(5).

The controller design is done to minimize the quadratic performance index

$$J(u) = \sum_{k=0}^{\infty} [x(k)^T Q x(k) + u(k)^T R u(k)], \tag{7}$$

where Q and R are positive definite matrices chosen to ensure acceptable transient response of the closed-loop system.

The optimal control which minimizes (7) in respect to the system (6) is given by [5]

$$u(k) = -Kx(k), \tag{8}$$

where the optimal feedback matrix K is determined by

$$K = (R + B^T PB)^{-1} B^T PA \tag{9}$$

and the matrix P is the positive definite solution of the discrete-time matrix algebraic Riccati equation

$$A^T PA - P - A^T PB(R + B^T PB)^{-1} B^T PA + Q = 0. \tag{10}$$

The matrix K is computed by the function `dlqr` of MATLAB®.

Let the matrix K be partitioned according to the dimensions of $x_{\dot\phi}$, $x_{\dot\Theta}$, Θ, $x_{\dot\phi_i}$, x_i as

$$K = [K_{\dot\phi}\ K_{\dot\Theta}\ K_\Theta\ K_{\dot\phi_i}\ K_{x_i}].$$

Since the state $x(k)$ of system (6) is not accessible, the optimal control (8) is implemented as

$$u(k) = -K_{\dot\phi}\hat{x}_{\dot\phi}(k) - K_{\dot\Theta}\hat{x}_{\dot\Theta}(k) - K_\Theta\hat\Theta(k) - K_{\dot\phi_i}\hat{x}_{\dot\phi_i}(k) - K_{x_i}\hat{x}_i(k), \tag{11}$$

where $\hat{x}_{\dot\phi}(k)$, $\hat{x}_{\dot\Theta}(k)$ are estimates of $x_{\dot\phi}(k)$ and $x_{\dot\Theta}(k)$, respectively, and

$$\begin{aligned} \hat{x}_{\dot\phi_i}(k+1) &= \hat{x}_{\dot\phi_i}(k) - T_0\hat{\dot\phi}(k), \\ \hat{x}_i(k+1) &= \hat{x}_i(k) + T_0(r_\Theta(k) - \hat\Theta(k)) \end{aligned} \tag{12}$$

are estimates of $x_{\dot\phi_i}(k)$ and $x_i(k)$, respectively. The estimates $\hat{\dot\phi}(k) = C_{\dot\phi}\hat{x}_{\dot\phi}(k)$ and $\hat\Theta(k)$ are obtained by a Kalman filter. The gyroscope under consideration contains a significant noise ϕ_g which is modeled by the additional equation

$$\dot\phi_g(k+1) = \dot\phi_g(k) + J_g\nu_g(k), \tag{13}$$

where ν_g is a white gaussian noise with unit variance and the coefficient J_g is determined experimentally to obtain a good estimate of $\dot\phi$ as $J_g = 0.0001$. Combining equations (1), (2), (3), and (13), the Kalman filter is designed for the system

$$x_f(k+1) = A_f x_f(k) + B_f u(k) + J_f\nu_f(k), \tag{14}$$

$$y_f(k) = C_f x_f(k) + H_f\nu_f(k), \tag{15}$$

where

$$x_f = [x_{\dot\phi}^T\ x_{\dot\Theta}^T\ \Theta\ \phi_g]^T, \quad y_f = [\dot\phi\ \dot\Theta\ \Theta\ \phi_g]^T, \quad \nu_f = [\nu_{\dot\phi}\ \nu_{\dot\Theta}\ \nu_g]^T$$

and A_f, B_f, J_f, C_f, H_f are matrices of corresponding dimensions.

A 17th order discrete-time Kalman filter for the system (14) is obtained as

$$\begin{aligned} \hat{x}_f(k+1) &= A_f\hat{x}_f(k) + B_f u(k) + K_f(y_f(k+1) - C_f B_f u(k) - C_f A_f \hat{x}_f(k)), \\ \hat{y}_f(k) &= C_f\hat{x}_f(k). \end{aligned} \tag{16}$$

The filter matrix K_f is determined as

$$K_f = D_f C_f^T (C_f D_f C_f^T + 10^{-5})^{-1}, \tag{17}$$

where the matrix D_f is obtained as the positive semi-definite solution of the discrete-time matrix algebraic Riccati equation

$$A_f D_f A_f^T - D_f - A_f D_f C_f^T (C_f D_f C_f^T + 10^{-5})^{-1} C_f D_f A_f^T + J_f D_{\nu_f} J_f^T = 0 \tag{18}$$

and the matrix $D_{\nu_f} = I_3$ is the variance of the noise ν_f. Note that in equations (17)), (18) the variance of the zero output noise in (16) is taken equal to 10^{-5} to avoid singularity of the corresponding matrix. The matrix K_f is computed by the function `kalman` of MATLAB®.

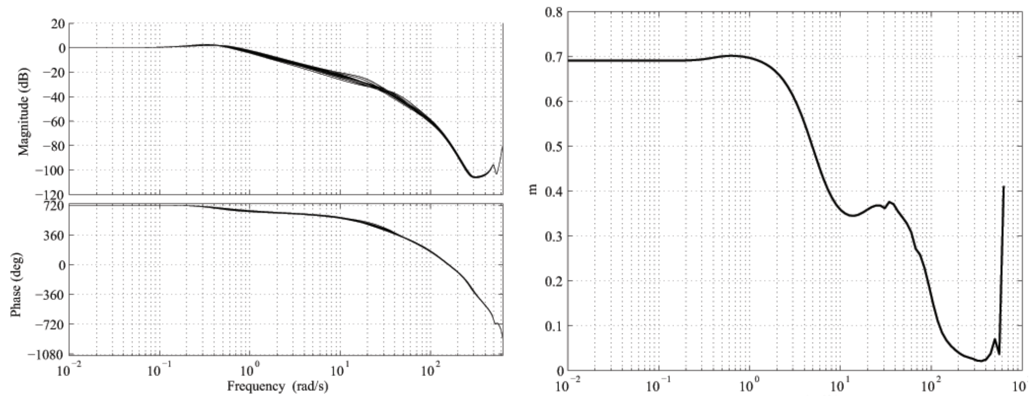
The quantity $\hat{\dot{\phi}}(k)$ which is used in computation of the estimate (12), is obtained from

$$\hat{\dot{\phi}}(k) = \dot{\phi}(k) - \hat{\dot{\phi}}_g(k), \tag{19}$$

where $\hat{\dot{\phi}}_g(k)$ is the last element of the state estimate $\hat{x}_f$.

The Bode plot of the closed-loop system obtained for the uncertain plant is shown in Fig. 2*a*. It is seen that the system tracks accurately references with frequencies up to 1 rad/s.

In Figure 2*b* we present the structured singular value (μ) [8] corresponding to the robust stability of the closed-loop system. Since the value of μ is less than



a) *b*)

Fig. 2. *a*) Bode plot of the uncertain closed-loop system, *b*) Robust stability of the system

one for all frequencies, the closed-loop system achieves robust stability for the plant uncertainty determined during the identification procedure.

A PI controller of the angular motion around the vertical axis is also designed. Since the yaw angular velocity $\dot{\psi}$ is measured by a gyroscope of the same type as the gyro used to measure the tilt rate, a second order Kalman is designed to produce sufficiently accurate estimate $\hat{\psi}$ of the yaw angle.

4. Experimental results. A simulation scheme of the control system and a specialized software in MATLAB®/Simulink®environment is developed to implement the control code. With the aid of Simulink Coder® [6] and Code Composer Studio®, a code is generated from this software which is embedded in the Texas Instruments Digital Signal Controller TMS320F28335 [7].

Several experiments with the controllers designed are performed and comparison with the simulation results is done. The experimental results obtained during the real-time robot control and the corresponding simulation results are given in Fig. 3.

It is seen from Fig. 3*a* and *b* that the wheels track accurately the reference and the robots keeps well its vertical position. The usage of $\hat{\psi}(k)$ instead of $\psi(k)$ ensures exact rotation of the robot around the vertical axis, while there is an increasing with the time bias in the measured value of $\psi(k)$ due to the integration of the gyro noise (Fig. 3*c*).

REFERENCES

[1] Segway Personal Transporters. Bedford, NH, 2012. `http://www.segway.com`

[2] Yamamoto Y. NXTway-GS (Self-Balanced Two-Wheeled Robot Controller Design). `http://www.mathworks.com/matlabcentral/fileexchange/`

[3] Double Robotics. Sunnyvale, CA, 2014. `http://www.doublerobotics.com`

[4] Chan R. P. M., K. A. Stol, C. R. Halkyard. Annual Reviews in Control, **37**, 2013, 89–103. `http://dx.doi.org/10.1016/j.arcontrol.2013.03.004`

[5] Goodwin G. C., S. F. Graebe, M. E. Salgado. Control System Design. Upper Saddle River, NJ, Prentice Hall, 2001, ISBN 9780139586538.

[6] The MathWorks, Inc. Simulink Coder. Natick, MA, 2013. Available from: `http://www.mathworks.com/products/datasheets/pdf/simulink-coder.pdf`

[7] Spectrum Digital, Inc. eZdspTMF28335 Technical Reference, 2007. Available from: `http://c2000.spectrumdigital.com/ezf28335/docs/ezdspf28335c_techref.pdf`

[8] Gu D.-W., P. H. Petkov, M. M. Konstantinov. Robust Control Design with MATLAB®, 2nd ed. London, Springer-Verlag, 2013.

Department of Systems and Control
Technical University of Sofia
8, St. Kliment Ohridski Blvd
1756 Sofia, Bulgaria
e-mail: `ts_slavov@tu-sofia.bg`