Journal of Marine Science and Technology

Volume 28 | Issue 5 Article 10

MOUSE TYPE BALLBOT IDENTIFICATION AND CONTROL USING A CONVEX-CONCAVE OPTIMIZATION

Sudchai Boonto
Department of Control Systems and Instrumentation Engineering, King Mongkut's University of Technology Thonburi, Bangkok, Thailand., sudchai.boo@kmutt.ac.th

Surapong Puychaisong
Department of Control Systems and Instrumentation Engineering, King Mongkut's University of Technology Thonburi, Bangkok, Thailand.

Journal of Marine Science and Technology, Vol. 28, No. 5, pp. 404-410 (2020)
DOI: 10.6119/JMST.202010_28(5).0010

MOUSE TYPE BALLBOT IDENTIFICATION AND CONTROL USING A CONVEX-CONCAVE OPTIMIZATION

Sudchai Boonto, and Surapong Puychaisong

Key words: convex-concave, identification, Ballbot, robust control, fixed-structure controller design.

ABSTRACT

This paper shows how to identify and control a mouse type Ballbot. The Ballbot is an unstable complex system. Using the first principle law to model the robot is not accurate enough. An identification method is proposed. The model is transformed to be MIMO state space model and using a convex-concave optimization to design a robust PI+phase-lead controller. Experiments on a design Ballbot are presented. It turns out that an identification model of Bollbot and a PI+phase-lead controller design framework is suitable for using to control a Ballbot.

I. INTRODUCTION

A Ballbot is an innovation of a robot that can operate in an indoor environment. We can apply this kind of mobile robot in a tight area. Many applications can use the Ballbot, such as industrial space, healthcare, and also office automation. Lauwers et al. introduced the first Ballbot (Lauwers et al., 2001). The system used an inverse mouse (IM) shape to drive the ball on the base. Another type of Ballbot is three omnidirectional wheels (OW), first introduced by Kuagai and Ochiai (2009). The OW type is more popular than the IM shape due to the IM type lacks yaw control. However, the OW type is more difficult to control than the IM type robot and requires omnidirectional wheels. The IM type robot has been improved to solve the drawback by adding a yaw drive (Kumaga and Ochiai, 2009). In the literature, the dynamic model of the Ballbots, both IM and OW, types are hard to derive even they are systematic. Not only the complexities to derive the dynamic equations in terms of the state-space equations, but each part of the robot also has to measure correctly, for example, each moment of inertia of each axis and the position of the center of gravity (Prieto et al., 2012; Bjärenstam and Lennartsson, 2012).

The most challenging part of designing the Ballbot is to construct a dynamic model. As shown in the literature, there is very complicated to derive a precise mathematic model of the Ballbot. The length and weight of each component have coursed uncertainty to the robot. Instead of obtaining the model of the robot by using the Euler-Lagrange equation, one can use an identification technique (Boonto and Werner, 2008). Using collecting input-output data to build a model, we do not need to measure the Ballbot parameters. Even the Ballbot is a nonlinear unstable open-loop system; we can trial-and-error tune with experiment a PID controller to stabilize the robot. Therefore, one could use a closed-loop system identification approach (Ljung, 1999) to build a mathematic model. Since the model is produced using real operating data of the robot, the model is very accurate and suitable for controller design.

In this work, we introduce a framework to identify and control a mouse type Ballbot. This paper organizes as follows. We describe the mechanical structure of the Ballbot in Section II. Section III explains the system identification approach. The procedure of how to design a H_∞ controller and a robust PI + phase-lead controller using the convex-concave optimization is shown in Section IV and V, respectively. The experimental result is shown in Section VI. Finally, conclusions are drawn in Section VII.

II. SYSTEM DESCRIPTION

Our Ballbot is a mouse type, as shown in Fig. 1, which is similar to CMU Ballbot(Nagarajan et al. 2014) but simpler. The robot included a NI MyRIO 1900 board and mid-range IMU sensor unit, four motor drivers, four rollers, and four DC motors. Every two motors control one direction and use only one control signal, e.g., the x-axis and y-axis direction. The robot is a height of 1.5 meters and weighs 15 kilograms. The ball is a 10.6 cm bowling ball, which is strong enough to support the robot. We control the robot by sampling the control signal at 10 msec, which is suitable for the controller board and the natural frequency of the robot. Fig. 2(a) and (b) show the structure of the design Ballbot. Each motor is attached to a

Paper submitted 01/21/20; revised 05/03/20; accepted 07/06/20. Corresponding Author: Sudchai Boonto (e-mail: sudchai.boo@kmutt.ac.th).
Department of Control Systems and Instrumentation Engineering, King Mongkut's University of Technology Thonburi, Bangkok, Thailand.

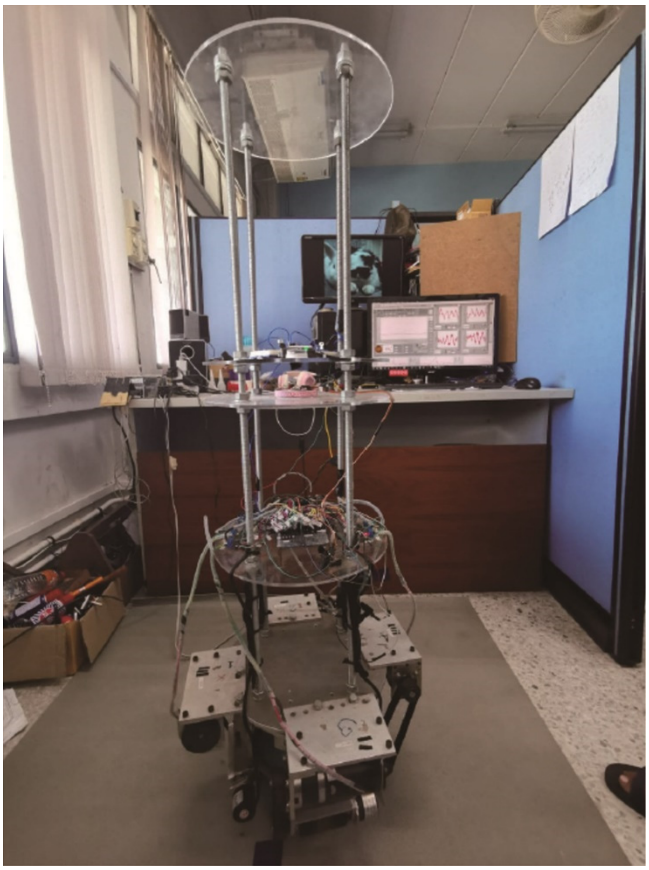
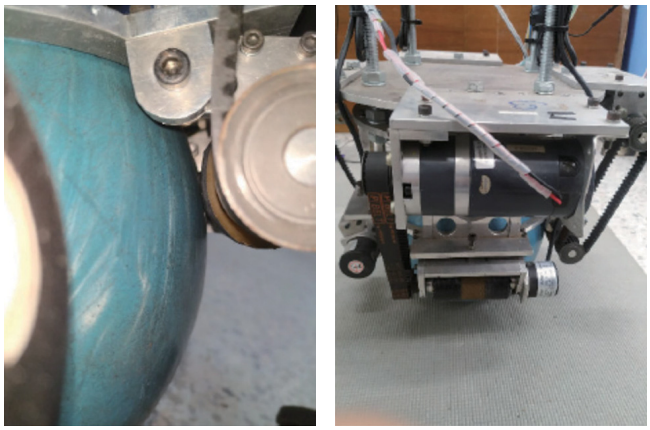


Fig. 1. The ballbot under closed-loop control



(a) (b)

Fig. 2. (a) ball and roller (b) the Ballbot on the floor

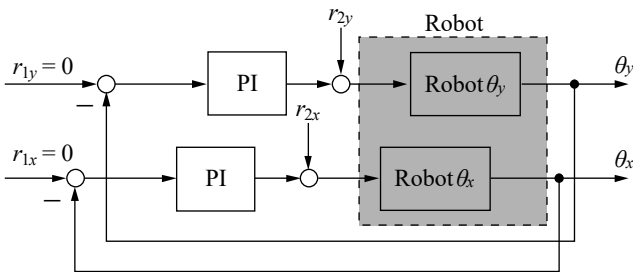


Fig. 3. A closed-loop system configuration for system identification

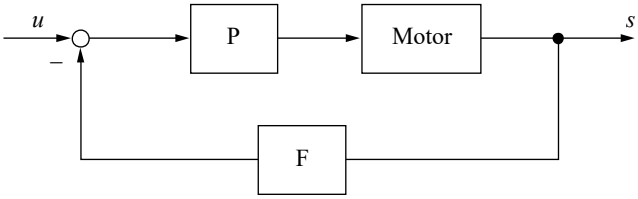


Fig. 4. Inner loop control

roller using a pulley and attached to the ball.

Even the robot looks reasonable to model with a standard method like the first principle method. However, it is still tough to find an accurate model, especially for controller design purposes. There are many uncertainties in each mechanical part, and this makes the model is not specific enough to simulate as the robot.

III. SYSTEM IDENTIFICATION

As mention in the previous section, it is difficult to derive the mathematical model of the Ballbot using the theoretical modeling method (Prieto, 2012). This work considers using a closed-loop identification framework (Ljung, 1999; Boonto, 2011; Thabthimratthana et al., 2016) to identify the robot model. Since the robot is an open-loop unstable system, it required an initial stabilized controller. We use two-loop control, as shown in Fig. 3. The outer loop controls the positions of the robot. The outer loop configuration contains two simple PI controllers for initially control. Since the structure of the hardware for both directions, x and y, are nearly identical, and then we can use PI controllers with the same parameters for both loops control. The shaded box in Fig. 3 is an inner loop of the control configuration. The inner loop controls the speed of DC motors, as shown in Fig. 4, which uses only a simple P controller for each loop, where F is a scaling gain to change the speed to round per minute (rpm) unit. Since the P controllers aim to control the speed of the motors, thus we can tune them by hand. Both motors are identical; only one tuning parameter for P control can be used. The P controller parameters are the crucial part because we need to use them to stabilize the unknown nonlinear system. We tuned the parameters by a trial-and-error technique and applying it to stabilize the robot at the reference point as long as possible. However, it requires some level of experience. Note that for the closed-loop system identification, the controller that is used to stabilize the system in the collecting data step must be as simple as possible. This requirement is to avoid the controller contaminate the feedback data.

The input-output data for identification are collected by applying two multisine excitation signals (Ljung, 1999) r_{2x} and r_{2y} to the robot that is holding at $(0,0)$ operating point. The sampling rate to obtain the data is 10 msec. Both excitation

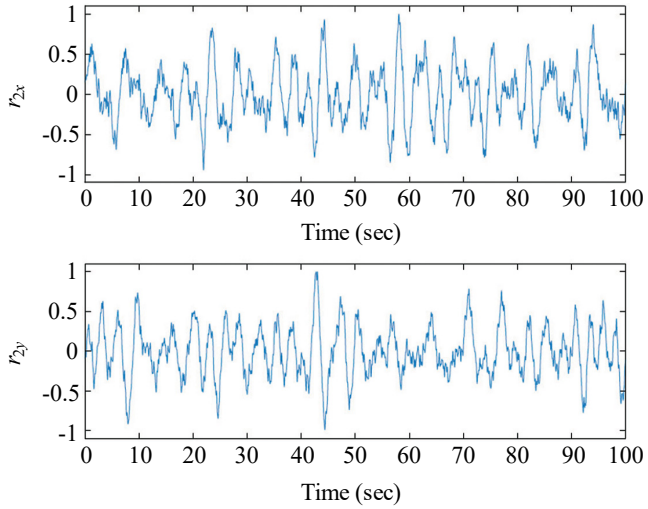


Fig. 5. Excitation signals

signals are separated into three parts, 0.1-0.4 Hz, 0.4-4 Hz, and 4-10 Hz. The amplitude of each part is 0.09 rpm, 0.03 rpm, and 0.016 rpm, respectively. We use three parts of the excitation signal to guarantee that the signals are rich enough for all frequency range. The amplitude of the high-frequency signal must be smaller than the signal at the low-frequency range. The robot cannot follow the high amplitude signal if it is high frequency. Moreover the signal r_{2x} and r_{2y} must uncorrelated. We use a random phase technique (Ljung, 1999) to prevent this phenomenon. The designed signals are shown in Fig. 5.

The input-output data are used to construct a two-input two-output ARX model with the system identification toolbox of MATLAB. Based on the inverse pendulum structure of the Ballbot, the dynamic model of the Ballbot described in (Lauwers et al., 2001) is four orders. In this work, we simply fixed the past output sample to have four samples and trial-and-error select the number of past input and delay. Finally, the model has four past sample outputs, three past sample inputs, and one delay sample for each input-output channel as follow:

$$A_{11}(z)y_1(k) = -A_{12}(z)y_2(k) + B_{11}(z)u_1(k) + B_{12}(z)u_2(k) + e_1(k) \quad (1)$$

$$A_{21}(z)y_2(k) = -A_{22}(z)y_1(k) + B_{21}(z)u_1(k) + B_{22}(z)u_2(k) + e_2(k) \quad (2)$$

, where

$$A_{11}(z) = 1 - 2.725z^{-1} + 2.54z^{-2} - 0.814z^{-3} - 0.0003993z^{-4}$$

$$A_{12}(z) = -0.2799z^{-1} + 0.5394z^{-2} - 0.2585z^{-3} - 0.002648z^{-4}$$

$$B_{11}(z) = -1.034 + 2.818z^{-1} - 2.628z^{-2} + 0.8436z^{-3}$$

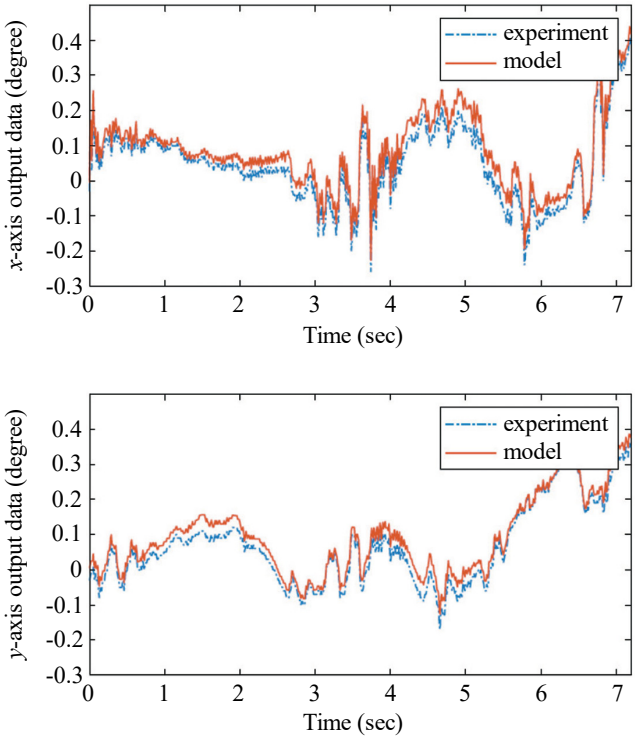


Fig. 6. Comparison between the closed-loop output signals of the model and measured data from the experiment.

$$B_{12}(z) = 0.001797 + 0.3116z^{-1} - 0.608z^{-2} + 0.2965z^{-3}$$

$$A_{21}(z) = 1 - 2.748z^{-1} + 2.553z^{-2} - 0.8063z^{-3} + 0.00194z^{-4}$$

$$A_{22}(z) = -0.1369z^{-1} + 0.2434z^{-2} - 0.106z^{-3} + 1.335\times10^{-5}z^{-4}$$

$$B_{21}(z) = 0.0006536 + 0.1394z^{-1} - 0.2489z^{-2} + 0.1085z^{-3}$$

$$B_{22}(z) = -1.126 + 3.092z^{-1} - 2.87z^{-2} + 0.924z^{-3}$$

$e_i(k)$ are white noise and z^{-1} here is a delay operator defined by $z^{-1}y(k) = y(k-1)$. The model is not only for open-loop simulation but also for controller design purposes. Hence the quality of the model is not only a similarity of the output of the model and the validation signal, but it must be controllable by the controller used in the identification phase. Here we manually adapt the model until the model is satisfied both criteria. To adjust the identified model, we change the condense of the excitation signal to match the frequency range around the crossover frequency. To check the quality of the model, we control the model with the same controller as we were collecting the data. The comparison of the closed-loop output of the model and measured data from the experiments are shown in Fig. 6. Note that from the frequency response of the system shown in Fig. 7, the phase of the system from input 1 to output

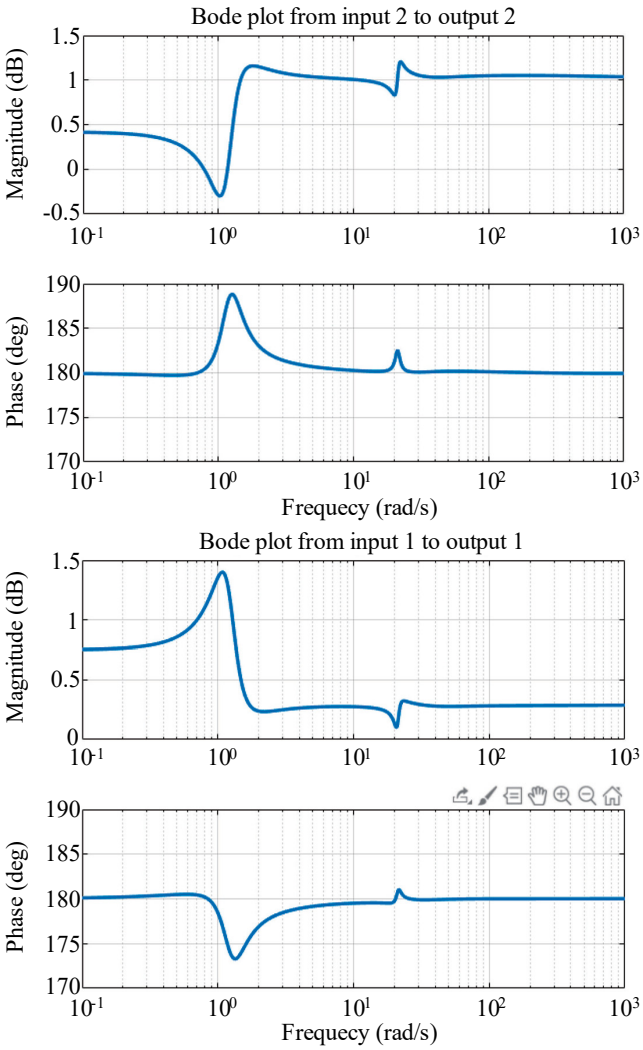


Fig. 7. The bode diagram of the identified model from input 1 to output 1 (upper), and from input 2 to output 2 (lower).

1 and from input 2 to output 2 are 180°. Then, it is clear that the input-output system is a non-minimum phase system.

IV. ROBUST CONTROLLER DESIGN

Since the identified model is built on the input-output data, which are in a linear range, the simple PID controller is not suitable to guarantee the high performance of the closed-loop system. The real robot is also an unstable nonlinear system. Here we want to control the Ballbot mostly in a vertical position, which is not far from the linear range. A robust controller can be used to design a controller that can deal with a model mismatch problem between the model and the real system. We create two robust controllers. One is a H_∞controller based on a mixed-sensitivity designed method, and the second one is a proposed robust PI + phase-lead controller. The robust PI + phase-lead controller is designed using a convex-concave optimization proposed by Hast et al (2013). Since the ARX

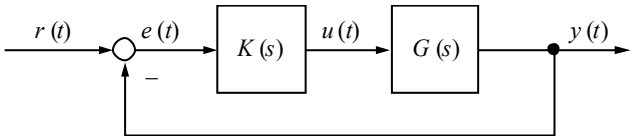


Fig. 8. Close-loop control configuration

model is a discrete-time model, it cannot directly develop the robust PI + phase-lead controller. We first transform the model into a continuous-time state-space model. For comparison purposes, the state-space model is the base model for designing both robust controllers.

The control configuration considered in this work is shown in Fig. 8. Also, the loop transfer function is $L(s) = K(s)G(s)$.

1. H_∞ controller

The H_∞controller is designed using a standard mixed-sensitivity method (Skogestad and Postlethwaite, 2005) with the hinfsyn command of MATLAB. We use two weighting filters W_s and W_k at the error signal and the control signal, respectively, the same for both input-output transfer functions. The weighting filters are

$$W_s(s) = \frac{0.5}{s + 5\times10^{-5}} \tag{3}$$

$$W_k(s) = \frac{2.5s + 2.5\times10^{-3}}{s + 0.1} \tag{4}$$

The resulted H_∞ controller is a 14th order controller. Unfortunately, as mention by Kirimi and Galdos (2011), in some cases, the full-order H_∞ controller cannot stabilize the system, including our case. This work the 14th order H_∞controller could not stabilize our real Ballbot. Then we reduce the order of the controller to be 4th order by using the ballreal command of MATLAB. The resulted H_∞ controller in transfer matrix form is

$$K_\infty(s) = \begin{bmatrix} K_{11}(s) & K_{12}(s) \\ K_{21}(s) & K_{22}(s) \end{bmatrix}, \tag{5}$$

Where

$$K_{11} = \frac{-1.272s^4 - 30.83s^3 - 586.5s^2 - 634.2s - 110.5}{s^4 + 62.15s^3 + 489.1s + 0.4541}$$

$$K_{12} = \frac{-0.3966s^4 + 21.45s^3 - 159.2s^2 - 133.8s - 16.01}{s^4 + 62.15s^3 + 489.1s + 0.4541}$$

$$K_{21} = \frac{0.05896s^4 + 45.49s^3 + 45.14s^2 - 26.65s - 54.69}{s^4 + 62.15s^3 + 489.1s + 0.4541}$$

$$K_{22}=\frac{-0.8815s^4-16.34s^3-713s^2-679.2s-8.532}{s^4+62.15s^3+489.1s+0.4541}$$

With this 4th order robust controller, the Ballbot can be controlled to stable at the origin without any problem.

Remark: In this paper, we aim to design the H_∞controller in order to assure that the identified model can be used to design an existing robust controller design technique. Here, the weighting functions (3) and (4) are not the best design function. The final performance of the controller can be improved depending on the experience of the user. For interested readers can consult (Gu et al. 2013) and reference therein.

V. FIXED-STRUCTURE CONTROLLER DESIGN USING A CONVEX-CONCAVE OPTIMIZATION

It is well known that the drawback of the mixed-sensitivity H_∞ design method is to give a high-order controller. In this research, we propose a fixed-structure robust controller design method using a convex-concave optimization technique (Hast et al. 2013). This framework was for a SISO system and extended to the MIMO system (Boyd et al. 2016). The following extension requires some more complicated mathematic. Our system is a MIMO system by the system in the x-axis, and the y-axis is nearly no correlation. Instead of using an intricate MIMO design, in this work, we use a SISO design for each input-output transfer function. The model is turned to be a decoupling model with standard similarly transformation (Antsaklis and Michel 2006). Then, we can design a SISO controller for each channel separately.

1. Robust PI + phase-lead controller structure

The fixed-structure controller used in this work is

$$K_{PI}(s)=K_P+\frac{K_I}{s}+K_d\frac{1+\frac{s}{b}}{1+\frac{s}{a}}. \tag{6}$$

There are five parameters to be turned, namely $K_P, K_I, K_D, a,$ and b. However, we can fix the phase-lead part the reduce the number of search parameters with a standard phase-lead design procedure (Qiu and Zhou, 2010).

A phase-lead term is significant for the controller because the phase-lead term does improve not only the damping ratio of the closed-loop system but also limits the gain of the controller at high-frequency. Parameters a and b in (6) can be selected according to the required adding phase-margin $\tilde{\phi}_m$ at the designed crossover frequency ω_c. By considering the Ballbot as a pendulum, the natural frequency of the robot is about 3.13 rad/sec. We selected the crossover frequency at least two times higher than the natural frequency at $\omega_c = 7$ rad/sec, then a and b can be determined as follows

$$\frac{a}{b}=\frac{1+\sin\tilde{\phi}_m}{1-\sin\tilde{\phi}_m} \tag{7}$$

$$\sqrt{ab}=\omega_c \tag{8}$$

Here the value of $\tilde{\phi}_m$ is 40^o, then we obtain a=14.4928 and b = 3.2642.

2. Optimization

Searching the optimal values of the parameters K_P, K_I, and K_D. We use a convex-concave procedure proposed by (Hast et al. 2013) for each input-output of the robot separately.

In this work, the constraint is a circle constraint (Hast et al. 2013). The stable closed-loop system must have $L(j\omega)$ in the Nyquist's diagram lines outside this circle with center c and radius r and equivalent to

$$r-\left|L(j\omega)-c\right|=r-g(\lambda)\le 0, \tag{9}$$

where $\lambda=\left[K_P\ K_I\ K_D\right]^T$ is a vector of the designing parameters. The control objective of this work is to minimize the Integrated Error (IE) of the controlled system. In (Åström and Hägglund, 2006), it has been shown that

$$\mathrm{IE}=\int_0^\infty e(t)\,dt=\frac{1}{K_I} \tag{10}$$

Then the minimize IE is equivalent to the maximize K_I as an optimization objective. Using a linearization around the frequency point, we obtain the optimization problem as follow:

$$\underset{K_P,K_I,K_D}{\text{maximize}}\quad K_I \tag{10}$$

$$\text{subject to}\qquad r-\mathfrak{Re}\left(\frac{\left(L_k(j\omega)-c\right)^*}{\left|L_k(j\omega)-c\right|}\left(L_k(j\omega)-c\right)\right), \tag{11}$$

Where $\mathfrak{Re}$ and $()^*$ denote the real part value and complex conjugate, respectively. Also, L_k is a loop-gain transfer function, and the subscript k denotes the iteration index of the optimization round.

3. Robustness Constraints

Since the model of the robot cannot cover all information of the robot, one has to define robust constraints carefully. In the circle criteria framework, there are two parameters are tunable, namely, M_s and M_t, which are defined as a $\|\mathrm{S(s)}\|_\infty$and $\|\mathrm{T(s)}\|_\infty$, respectively. Here $S(s)$ is a sensitivity function, and $T(s)$ is a complementary sensitivity function (Skogestad and Postlethwaite 2005), as follows:

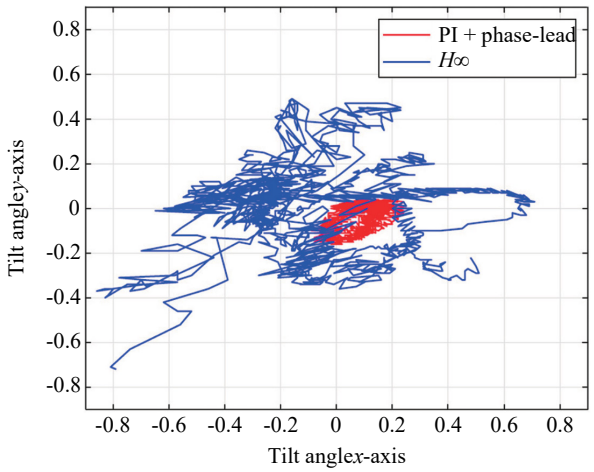


Fig. 9. Control results of PI+phase-lead controller and reduced order H_∞ controller.

$$S(s)=\frac{1}{1+L(s)},\ T(s)=\frac{L(s)}{1+L(s)} \tag{12}$$

For the robustness of the system, the sensitivity function indicates that $L(j\omega)$ should be outside two circles on the Nyquist's diagram (Doyle et al. 1990). One circle centers at c_s=–1 and radial $r_s=1/M_s$ and a second circle centers at $c_t=-M_t^2/(M_t^2-1)$ and radius $r_t=-M_t/(M_t^2-1)$.

4. Resulted in Robust PI + phase-lead controller

With a and b parameters designed above, and the initial values K_P=22, K_I=3, and K_D= -11 for both input-output channels. Then, the initial PI + phase-lead controller is

$$K_{PI}(s)=22+\frac{3}{s}-11\frac{1+\frac{s}{14.4928}}{1+\frac{s}{3.2642}} \tag{13}$$

The original optimization problem is infinite in terms of the frequency point. The frequency range is gridding of 1000 points from 0.07 Hz to 100 Hz in the logarithm scale to make it finite. The optimal problem is solved using convex optimization with the CVX package (Grant and Boyd 2014) on MATLAB. The results controller for x-axis and y-axis are

$$K_{PI}^{x}(s)=0.5229+\frac{15.0961}{s}-0.0168\frac{1+\frac{s}{14.4928}}{1+\frac{s}{3.2642}} \tag{14}$$

$$K_{PI}^{y}(s)=0.4909+\frac{15.0061}{s}-0.0144\frac{1+\frac{s}{14.4928}}{1+\frac{s}{3.2642}}. \tag{15}$$

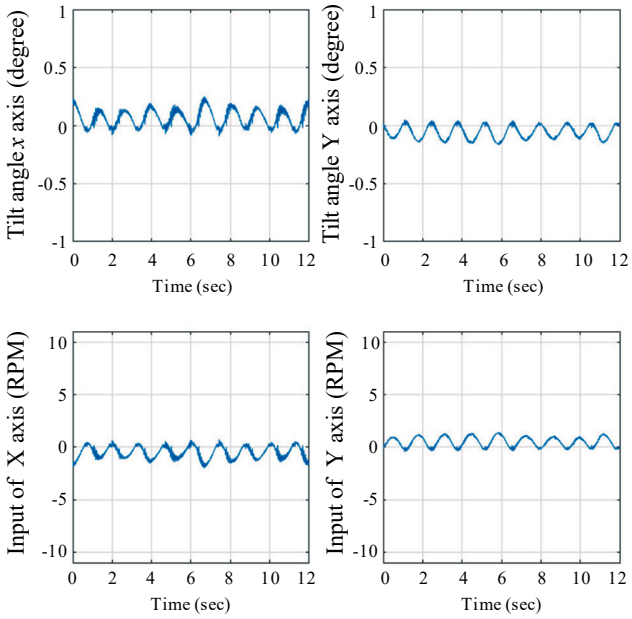


Fig. 10. The time-domain plots of the closed-loop systems with the PI+phase-lead controller : degree (above), RPM (below).

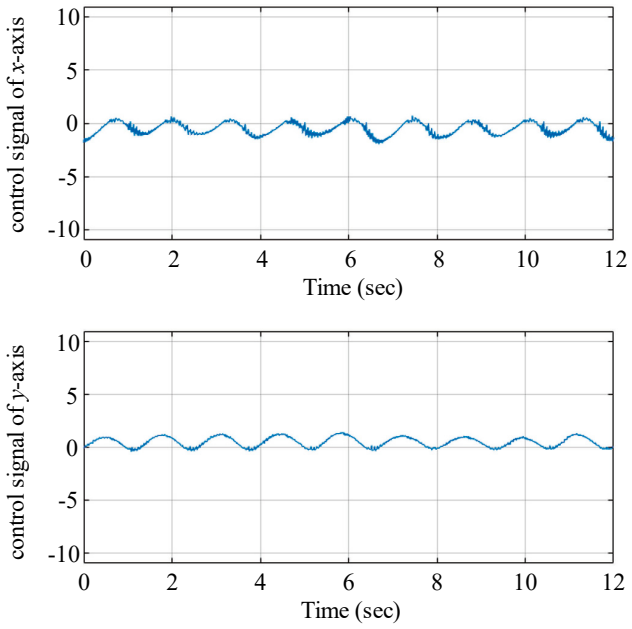


Fig. 11. The time-domain plots of the control signal of the closed-loop system with the PI+phase-lead controller : $x-$ axis (above), $y-$ axis (below).

VI. EXPERIMENTAL RESULTS AND DISCUSSION

The controllers are applied to control the Ballbot on the MyRIO board under LabVIEW real-time environment. The sampling rate is set at 10 msec. Two motors share the same control input for the x-axis, and the other two motors share the same control input for the y-axis. The control objective is to

stabilize the robot at the (0,0) position. As shown in Fig. 9, both controllers can stabilize the Ballbot at the reference position. The error is not higher than $\pm 0.2^o$. The time-domain plots of the closed-loop system in terms of degree and RPM are shown in Fig. 10 for the closed-loop with the PI+phase-lead control. The closed-loop robot operates smoothly. The controller signals of both the x-axis and y-axis are sinusoidal shapes, reacting to the output of that axis, as shown in Fig. 11.

In terms of implementation, the PI+phase-lead is comfortable to apply directly to the LabVIEW environment on MyRIO board because the parameters of the control can fill in the standard block. So this control structure is suitable for most of the hardware systems not only in the lab-scale but also can be applied to implement on the industrial controller equipment like programmable logic controller (PLC). Moreover, since the controller has a low-order configuration, it has less numerical issues than the high-order controller. Even though we can reduce the order of the high-order controller, but it reduces the performance of the controller.

VII. CONCLUSION

In this work, we show how to use the identification framework and the fixed structure robust PI+phase-lead controller design with a convex-concave optimization to control a Ballbot. The method applies to a MIMO Ballbot system. By experiments, it turns out that the identification model of Bollbot and a PI+phase-lead controller design framework is suitable for using to control a Ballbot

REFERENCES

Antsklis, P. J. and A. N. Michel (2006). Linear Systems. Birkhäuser.

Åström, K. J. and T. Hägglund (2006). Advanced PID Control, ISA.

Bjärenstam, M. J. and M. J. Lennartsson (2012). Development of a ball-balancing robot with Omni-wheels. Master's thesis, Lund University, Department of Automatic Control.

Boonto, S. (2011). Identification of linear parameter-varying input-output models. Ph.D. dissertation, Institute of Control Systems, Technische Universität Hamburg-Harburg.

Boonto, S. and H. Werner (2008). Closed-loop system identification of LPV input-output models – application to an arm-driven inverted pendulum. Proceeding of the 47th IEEE Conference on Decision and Control, Cancun, Mexico, 2606-2611.

Boyd, S., M. Hast and K. J. Åström (2016). MIMO PID tuning via iterated LMI restriction. International Journal of Robust and Nonlinear Control, 26(8), 1718-1731

Doyle, J. C., B. Francis and A. Tannenbaum (1990). Feedback Control Theory. Macmillan Publishing Co.

Grant, M. and S. Boyd (2014). CVX: Matlab software for disciplined convex programming, version 2.1, http://cvxr.com/cvx.

Gu, D.-W., P. H. Petkov and M. M. Konstantinov (2013). Robust Control Design with MATLAB. 2nd edition, Springer London

Hast, M., K. J. Åström, B. Bernhardsson and S. Boyd (2013). PID design by convex-concave optimization. Proceeding of the 2013 European Control Conference (ECC), Zurich, Switzerland, 4450-4465.

Karimi, A. and G. Galdos (2010). Fixed-order H_∞ controller design for nonparametric models by convex optimization. Automatica, 45(8), 1388-1394

Kumaga, M. and T. Ochiai (2009). Development of a robot balanced on a ball — Application of passive motion to transport. Proceeding of IEEE International Conference on Robotics and Automation, Kobe International Conference Center, Kobe, Japan, 4106-4111.

Lauwers, T., G. Kantor and R. L. Hollis (2005). One is enough! Proceeding of the 12th International Symposium Robotics Research, San Francisco, CA, USA, 1781-1785.

Ljung, L. (1999). System Identification Theory for the User. 2nd edition. Prentice-Hall, New Jersey

Nagarajan, U., G. Kantor and R. Hollis (2014). The ballbot: An omnidirectional balancing mobile robot. The International Journal of Robotics Research, 33, 917-930.

Prieto, S. S., T. A. Navarro, M. G. Plaza and O. R. Polo (2012). A monoball robot based on LEGO Mindstorms. IEEE Control Systems Magazine, 32(2), 71-83.

Qiu, L. and K. Zhou (2010). Introduction to Feedback Control. Pearson.

Skogestad, S. and I. Postlethwaite (2005). Multivariable Feedback Control: Analysis and Design, 2nd edition, John Wiley & Sons

Thabthimratthana, C., S. Saelim, S. Tiewcharoen and S. Boonto (2016). Robust PID controller design using convex-concave optimization: Application to an unstable system. Proceeding of the 16th International Conference on Control, Automation and Systems, HICO, Gyeongju, Korea, 638-643.