Mechatronics 35 (2016) 102–108

Contents lists available at ScienceDirect

Mechatronics

journal homepage: www.elsevier.com/locate/mechatronics

Automatic calibration with robust control of a six DoF mechatronic system

Amélie Chevalier*, Cosmin Copot, Clara M. Ionescu, Robin De Keyser

Department of Electrical energy, Systems and Automation, Ghent University, Technologiepark 914, 9052 Gent, Belgium

ARTICLE INFO

Article history:
Received 21 October 2015
Accepted 13 January 2016
Available online 9 February 2016

Keywords:
Ball and plate
Stewart platform
Automatic calibration
Robust control
Mechatronics

ABSTRACT

This paper presents a robust control methodology incorporating an automatic calibration step to compensate dynamic variations in a 6 DoF mechatronic system. The application used to illustrate the efficacy of the proposed approach is the ball and plate system based on the Stewart platform. To emulate dynamic changes in the system, we make use of various types of balls, varying in mass, diameter and surface. An automatic calibration step is introduced to obtain a model based on the type of ball which is placed on the plate. Using a model-based tuning technique, an optimal proportional-differential (PD) controller is designed based on the previous calibration. The resulting controller is tested for robustness by changing the ball without changing the controller parameters. The results indicate that our setup is robust and performs well in the presence of significant changes.

1. Introduction

The ball and plate system is a typical benchmark of an open loop unstable system. It stirs up both scholar and commercial interest, as the real and potential applications are numerous. From the point-of-view of control theory, the challenge that this system presents makes it attractive for testing and developing new control approaches.

The ball and plate system is based on the Stewart platform published by Stewart in 1965 [1]. Since then, it attracted the attention of professionals in the discipline of electromechanical engineering [2]. The advantages of parallel manipulators over serial ones and its simple design makes the Stewart Platform a very broadly examined and utilized solution [3–6]. The advantages of such systems from dynamics and control point of view have been thoroughly examined and reported in the works of Wu [7–9].

Some examples of its application can be found today in spacecraft, aeronautics, entertainment or even medicine. The aerospace docking mechanism used to attach a space craft to the international space station is based on the Stewart platform [7,8]. The mechanism is designed to absorb the relative motion of the spacecraft. In aerospace training a traditional application of the Stewart platform is found under the form of a full flight simulator [9]. The ability of this device to move and rotate using its full six degrees of freedom, together with its strength and robustness makes of it a great candidate for this sort of tasks. It accurately reproduces the accelerations to which the crew will be subject during real maneuvers aboard an aerial vehicle. Also in medicine the Stewart platform is used for precision surgery [10,11]. Surgery has become an extremely precise practice and professionals often have to make movements in the order of hundreds of microns. The inherent limits of human dexterity are a constraint for new advances in this field. In this context, high-precision robotics are being developed and implemented for medical use. From the observation of these devices it is possible to acquire an idea of their versatility.

In these real applications of the Stewart Platform, the characteristics of the system may be subject to change (e.g. the mass fixed to the platform in a flight simulator changes with its occupancy, the inertia of the vessels to be docked change, the weight of the pilots in the flight simulator is subject to change). The controller, however, has to keep its performance constantly optimal, i.e. a robust control strategy is needed and changes to the system should be automatically detected.

In the current paper, we propose a fully automatic process, able to detect changes in the parameters of the ball and plate system to trigger its identification and, subsequently, to perform the parametric optimization of its robust proportional-differential (PD) controller. The application incorporates vision based feedback which is dependent on the environmental varying light conditions. Under such conditions, adaptations are carried out and triggered automatically whenever necessary. The robustness of the controller

* Corresponding author. Tel.: +32 92645580.
E-mail addresses: Amelie.Chevalier@UGent.be (A. Chevalier), Cosmin.Copot@UGent.be (C. Copot), ClaraMihaela.Ionescu@UGent.be (C.M. Ionescu), Robain.DeKeyser@UGent.be (R. De Keyser).

http://dx.doi.org/10.1016/j.mechatronics.2016.01.005
0957-4158/

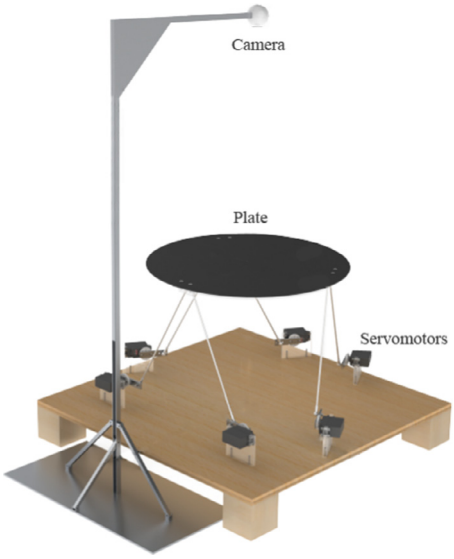


Fig. 1. The ball and plate laboratory set-up.

Table 1
Summary of the characteristics of the balls used for testing the system.

Ball ID	Diameter (mm)	Weight (g)
Pétanque ball	75	718
White-gray poolball	56	246
Hockey ball	73	152
Beige poolball	44	116
Ping-pong ball	40	2.7

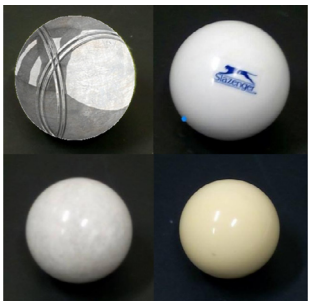
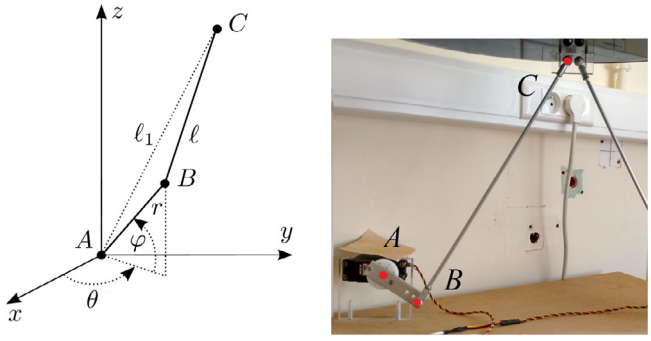


Fig. 2. 4 different types of balls: Pétanque ball (top left), White-gray poolball (bottom left), beige poolball (bottom right), Hockey ball (top right). (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)



(a) Schematic representation of the mechanism. (b) Real appearance of the mechanism.

Fig. 3. Illustration of the servomotor mechanism.

obtained through model-based tuning is tested in a set of measurements using different types of balls with a wide range of weights, diameters and surfaces.

This paper is structured as follows: the next section gives a description of the ball and plate system used in this study. Section 3 discusses the automatic calibration step. This step is based on a mathematical model which is firstly described after which the automatic calibration is presented. Section 4 presents the model-based tuning procedure of the PD controller. Section 5 presents the measurements and the corresponding results. A conclusion is formed in Section 6.

2. System description

The ball and plate system, just like its two-dimensional counterpart (i.e. the ball and beam system) is an open loop unstable system. It consists of four main parts: the plate, the ball, the feedback sensor and the computer control unit. The complete system can be seen in Fig. 1.

The *Plate* is embodied by a Stewart Platform. Its circular surface (commonly referred as 'platform'), with a diameter of 575 mm, is covered with a black foil to enhance the visual contrast between the ball and the surface of the plate. The plate is actuated by six angular servomotors, which determine the position and inclination of the platform by a connection using stiff rods and ball joints between the servomotors and the plate. The positions of the servomotors are governed by a micro-controller, which receives the input signal for the angle from the computer.

There are five different *ball* types used in this research, a ping-pong ball and other 4 balls as in Fig. 2. Their characteristics are shown in Table 1. Note that every type of ball has a different mass and diameter, which will be used to test the robustness of the controller.

As *sensor*, a small, low-cost camera placed above the platform is used to capture the position of the ball. Its data is streamed to the computer via USB connection.

All the calculations are made digitally by the *computer*, which collects the data from the camera, analyzes it and sends the required angular positions to the micro-controller based on the desired reference trajectory. This trajectory is defined by the user and represents the *X*–*Y* positions of the ball on the plate and can be uploaded as a TXT file. For instance, the center of the plate is position 0–0. The program running in the computer integrates several different features, including computer vision (e.g. Hough transform).

A detailed information about the software and hardware used for this setup has been reported in [12] and in [13].

3. Automatic system calibration

The very first step in understanding and controlling the system is establishing a computable relation between the output (i.e. the position of the ball) and the input (i.e. the voltages send to the servomotors). This is achieved as the result of an automatic calibration step based on insight into the mathematical modeling.

3.1. Mathematical modeling

The modeling of the system consists of three main problems regarding the relationships between: i) the servomotor angles versus the coordinates of the junction points (see Fig. 3), ii) the

coordinates of the junction points versus the inclination of the platform, and iii) the inclination of the platform versus the position of the ball. The first two are static problems, which involve mainly geometrics while the third problem requires an analysis of the dynamics of the ball and plate system.

Notice that this specific implementation and coordinate system of the ball and plate system will simplify the control problem since it involves the coordinates for position only. Specifically, speed and acceleration are not considered in this work.

3.1.1. Servomotor angles versus the coordinates of the junction points

Using the notation in Fig. 3, C is the junction point, B is the intermediate joint and A is the axis of the servomotor, φ is the servomotor angle to be calculated (i.e. the shaft angle of the servomotor).

The point B can be expressed with respect to the main coordinate system as:

$$\begin{cases} x_B = r\cos(\varphi)\cos(\theta) + x_A \\ y_B = r\cos(\varphi)\sin(\theta) + y_A \\ z_B = r\sin(\varphi) + z_A \end{cases} \tag{1}$$

where the definition of r, ϕ, θ can be found in Fig. 3a. Based on geometric relationships in Fig. 3a, the following equations can be obtained:

$$\begin{cases} (x_C - x_A)^2 + (y_C - y_A)^2 + (z_C + z_A)^2 = l_1^2 \\ (x_B - x_A)^2 + (y_B - y_A)^2 + (z_B + z_A)^2 = r^2 \\ (x_C - x_B)^2 + (y_C - y_B)^2 + (z_C + z_B)^2 = l^2 \end{cases} \tag{2}$$

Combining Eqs. (1) and (2), the system can be solved for φ:

$$\varphi = \arccos\left(\frac{M}{\sqrt{K^2 + L^2}}\right) + \arctan\left(\frac{L}{K}\right) + \pi \tag{3}$$

with

$$\begin{aligned} K &= 2(x_A - x_C)r\cos(\theta) + 2(y_A - y_C)r\sin(\theta) + y_A + x_A \\ L &= 2(z_A - z_C)r \\ M &= l^2 - l_1^2 - r^2 \end{aligned} \tag{4}$$

With this result in combination with the transformation matrix previously derived, the angle of the servomotor can be calculated from the inclination of the platform. Note that this method will be applied six times, once for each of the servomotors.

3.1.2. Coordinates of the junction points versus the inclination of the platform

In order to calculate the relationship between a desired inclination angle of the platform and the position of the junction points, a transformation matrix is needed. Homogeneous coordinates are used throughout this step.

The transformation matrix for a rotation around an arbitrary axis (x and y axis of the platform, in our case) is compound of the translation matrix from the axis to the origin, T_1, the rotation matrix about said axis, T_2, and the translation matrix from the origin back to the previous position of the axis, T_3 [14–16]. Fig. 4 illustrates these three transformations. The combination of these transformation matrices gives:

$$C_3 = T\ C_0 = T_3\ T_2\ T_1\ C_0 \tag{5}$$

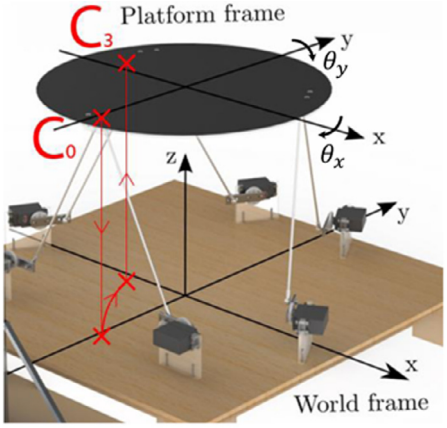


Fig. 4. Representation of the three transformations: T1, T2, T3. C_0: neutral position of the junction; C_3: position of the junction after transformation.

with

$$T_1 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -h \\ 0 & 0 & 0 & 1 \end{bmatrix};\ T_2 = \begin{bmatrix} \vec{e_x} & \vec{e_y} & \vec{e_z} & \vec{0}_{3x1} \\ 0 & 0 & 0 & 1 \end{bmatrix};$$
$$T_3 = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & h + \Delta z \\ 0 & 0 & 0 & 1 \end{bmatrix};\ \vec{e_x} = \begin{bmatrix} \cos(\theta_x)\cos(\Delta\theta_z) \\ \cos(\theta_x)\sin(\Delta\theta_z) \\ \sin(\theta_x) \end{bmatrix}; \tag{6}$$
$$\vec{e_y} = \begin{bmatrix} \cos(\theta_y)\sin(\Delta\theta_z) \\ \cos(\theta_y)\cos(\Delta\theta_z) \\ \sin(\theta_y) \end{bmatrix};\ \vec{e_z} = \vec{e_x} \times \vec{e_y}$$

where h is the height of the platform, Δz is the translation along the z axis (optimally, −46 mm [12]), θ_x, θ_y are the desired inclination angles of the platform, $\vec{0}_{3x1}$ is a 3 by 1 vector with all elements 0 and $\Delta\theta_z$ is the undesired equivalent rotation about the z axis which can be calculated as:

$$\Delta\theta_z = \frac{\arcsin(\tan\theta_x\ \tan\theta_y)}{2} \tag{7}$$

3.1.3. Inclination of the platform versus the position of the ball

The third part of the model is a dynamic model describing the movement of the ball on the plate. As mentioned before, this is a nonlinear system. However, a linear theoretical model will be derived, for which some preliminary assumptions will be made:

- The ball rolls without slipping.
- The ball is at each moment in contact with the platform.
- Friction is proportional to the velocity of the ball and uniform in both directions.
- Rotational energy of the ball due to rotation of the platform is negligible.
- θ_x, θ_y are close to zero, thus allowing for small-angle approximations.

The Euler–Lagrange method is used to calculate the differential equation governing the system:

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} = Q \tag{8}$$

where q represents the general coordinates, L is the Lagrangian and Q represents the external forces. These equations result, after simplification and linearization, in:

$$\left(m + \frac{J}{R^2}\right)\ddot{x} + mg\theta_x + b\dot{x} = 0 \tag{9}$$

$$\left(m + \frac{J}{R^2}\right)\ddot{y} + mg\theta_y + b\dot{y} = 0 \tag{10}$$

where J represents the moment of inertia of the ball, R represents the radius of the ball, m is the mass of the ball, b represents the friction coefficient and g is the gravitational constant. Using the Laplace transform on these equations allows for the derivation of the system's transfer function.

$$\frac{X(s)}{\Theta_x(s)} = -\frac{mg}{(m + \frac{J}{R^2})s^2 + bs} \tag{11}$$

Note only the equation for the x-axis is explicitly expressed; as the system is symmetric, the equation of the y-axis is identical. Taking into account that the moment of inertia for spherical objects can be expressed as $J = kmR^2$ [13], Eq. (11) becomes:

$$\frac{X(s)}{\Theta_x(s)} = -\frac{\frac{g}{1+k}}{s(s + \frac{b}{m(1+k)})} = \frac{K}{s(s+a)} \tag{12}$$

It now becomes evident that the transfer function only depends on the coefficients k, b and m, all of them characteristics of the ball.

The model from (12) is identified from real data from the ball and plate system. The identification procedure is considered a calibration procedure and described later in Section 5.

3.2. Calibration procedure

Based on the mathematical model derived in previous section, an automatic calibration step is developed for the ball and plate system. As can be seen in the mathematical model, the system's dynamics depend on several features of the ball (e.g. mass, inertia). Taking into account the necessity of the system to adapt to this variable scenario, a method must be implemented such that the coefficients of the transfer function can be recalculated simultaneously with the normal operation of the system (i.e. on-line). This will allow for the controller to be tuned to any new ball used on the platform.

The automatic calibration step consists of an on-line identification of the process. Therefore, the generalized model for discrete systems ARMAX is chosen and represented by:

$$\begin{aligned} &y_i + a_1 y_{i-1} + \ldots + a_{n_a} y_{i-n_a} \\ &\quad = b_1 u_{i-1} + \ldots + b_{n_b} u_{i-n_b} + c_1 e_{i-1} + \ldots + c_{n_c} e_{i-n_c} + e_i \end{aligned} \tag{13}$$

which can be shortly rewritten as:

$$y_i = \boldsymbol{\varphi}_i^T \boldsymbol{\theta} \tag{14}$$

with

$$\begin{aligned} \boldsymbol{\varphi}_i &= [-y_{i-1} \ \ldots \ -y_{i-n_a} \ u_{i-1} \ \ldots \ u_{i-n_b} \ e_{i-1} \ \ldots \ e_{i-n_c} \ e_i]^T \\ \boldsymbol{\theta} &= [a_1 \quad \ldots \quad a_{n_a} \quad b_1 \quad \ldots \quad b_{n_b} \quad c_1 \quad \ldots \quad c_{n_c} \quad 1]^T \end{aligned} \tag{15}$$

where y is the output of the system, u is the input of the system, e is the error caused by disturbances, and $\boldsymbol{\theta}$ is the vector of parameters to be identified. The transfer function of the process contains one integrator, which can corrupt the identification data and, therefore, it needs to be eliminated beforehand [17]. In order to remove the integrator from the identification process, the velocity of the ball, V_x is used as output data (i.e. y) instead of the position of the ball. θ_x is used as input data (i.e. u). Note that only data for the x-direction is used as the system is symmetric w.r.t. the y-direction.

The least squares (LS) method is chosen as identification method for its simplicity and low computational effort [18]. The equations to identify the parameter vector, $\boldsymbol{\theta}$, can be derived based on a set of j measurements which are available. Starting from Eq. (14), the following relationship can be written:

$$\begin{bmatrix} y_1 \\ y_2 \\ \vdots \\ y_k \end{bmatrix} = \begin{bmatrix} \varphi_{11} & \varphi_{12} & \ldots & \varphi_{1n} \\ \varphi_{21} & \varphi_{22} & \ldots & \varphi_{2n} \\ \vdots & \vdots & \ddots & \vdots \\ \varphi_{j1} & \varphi_{j2} & \ldots & \varphi_{j3} \end{bmatrix} \begin{Bmatrix} \theta_1 \\ \theta_2 \\ \vdots \\ \theta_n \end{Bmatrix} \tag{16}$$

Eq. (16) can be rewritten in matrix form as:

$$\boldsymbol{y} = \boldsymbol{\Phi}^T \boldsymbol{\theta} \tag{17}$$

By definition, the cost function to be minimized by the least squares method is:

$$J_{cost} = [\boldsymbol{y} - \boldsymbol{\Phi}^T \hat{\boldsymbol{\theta}}]^T [\boldsymbol{y} - \boldsymbol{\Phi}^T \hat{\boldsymbol{\theta}}] \tag{18}$$

The optimal $\hat{\boldsymbol{\theta}}$ is found by taking the partial derivative of the cost function to $\hat{\boldsymbol{\theta}}$ and solving the result for $\hat{\boldsymbol{\theta}}$. The obtained expression for $\hat{\boldsymbol{\theta}}$ is then:

$$\hat{\boldsymbol{\theta}} = (\boldsymbol{\Phi}\boldsymbol{\Phi}^T)^{-1}\boldsymbol{\Phi}\boldsymbol{y} \tag{19}$$

4. Robust PD control

As afore mentioned, the ball and plate system is an integrating system, and thus, requires feedback stabilization. In this case, the system is subject to change (i.e. replacement of the ball). For this reason, as already explained, the controller must be able to cope with the varying parameters of the system.

Among the different control strategies available, model-based tuning of proportional-derivative-integral (PID) controllers stands out and is chosen in this research for its simplicity and easy implementation. Future improvements can be done by adding the possibility to replace it by model-predictive control. Due to the already integrative behavior of the system, a PD controller suffices. The block diagram of the model-based tuning method for the PD controller is shown in Fig. 5.

Once the parameters of the system have been identified using the method described in previous section, we have all the information needed to completely define the system and, therefore, to derive the optimal controller values in a model-based tuning step.

A numerical method for optimization is required, as the equations involved in calculating the optimal controller are impractical to be solved analytically. Therefore, the function to be optimized and a set of constraints must be defined.

Starting from the equation of the PD controller, $C(s)$, and the transfer function of the plant, $G(s)$, (see (12)) the open loop transfer function, $L(s)$, can be calculated as:

$$L(s) = C(s)G(s) = (K_p + sK_d)\frac{K}{s(s+a)} \tag{20}$$

Subsequently, the sensitivity function, $S(s)$, and the closed-loop transfer function, $T(s)$, have the following expressions:

$$\begin{aligned} S(s) &= \frac{1}{1+L(s)} \\ T(s) &= \frac{L}{1+L} = \frac{KK_p + sKK_d}{s^2 + (a + KK_d)s + KK_p} \end{aligned} \tag{21}$$

To achieve a good transient response, the maximum overshoot, M_p, and the settling time, T_s, are constraint to certain limits (M_p^+ and T_s^+ respectively). The relation between maximum overshoot and settling time and the transfer function is expressed through (22) and (23), respectively [19].

$$M_p = \frac{1}{2\zeta\sqrt{1-\zeta^2}} \leq M_p^+ \tag{22}$$

$$T_s = \frac{4}{\zeta\omega_n} \leq T_s^+ \tag{23}$$

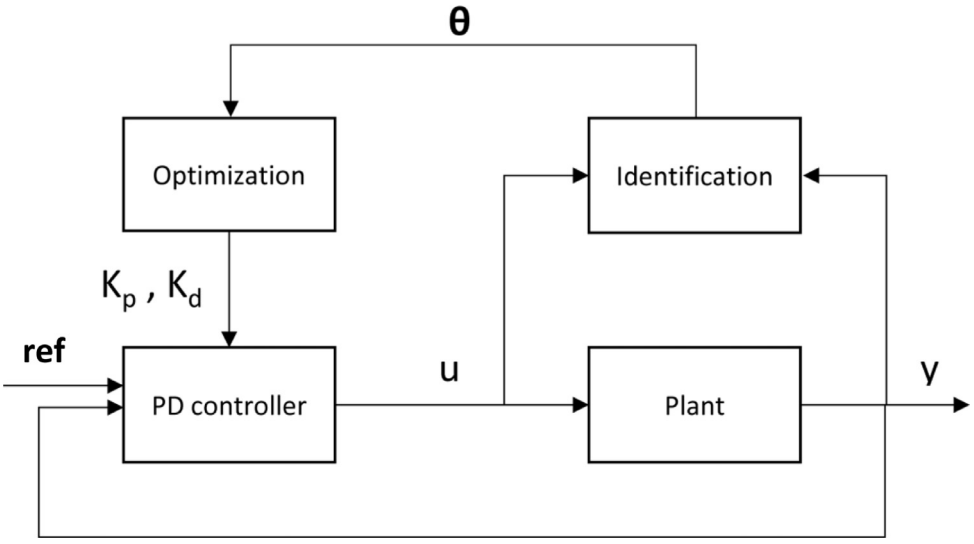


Fig. 5. Block diagram representation of the model-based tuning method for the PD controller.

where the characteristic polynomial of the closed-loop function is:

$$R(s) = s^2 + 2\zeta\omega_n s + \omega_n^2 \tag{24}$$

Equivalence of the terms in (24) with the values of the closed-loop transfer function in (21) yields:

$$\begin{cases} K_p = \dfrac{\omega_n^2}{K} \\ K_d = \dfrac{2\zeta\omega_n - a}{K} \end{cases} \tag{25}$$

From classical control theory, it is known that the sensitivity function, $S(s)$, represents a good estimation of the stability margin and the robustness of the closed-loop system [20]. Therefore, the inverse of the sensitivity function will be maximized resulting in the following set of equations:

$$\begin{cases} \max_\omega \mid 1 + L(j\omega) \mid \\ L(s) = \dfrac{K(K_p + sK_d)}{s(s+a)} \\ K_p = \dfrac{\omega_n^2}{K} \\ K_d = \dfrac{2\zeta\omega_n - a}{K} \\ \dfrac{1}{2\zeta\sqrt{1-\zeta^2}} \le M_p^+ \\ \dfrac{4}{\zeta\omega_n} \le T_s^+ \end{cases} \tag{26}$$

In this research the values for maximum overshoot and settling time are chosen to be $M_p^+ = 20\%$ and $T_s^+ = 2$ s. Using these values the optimal values for K_p and K_d for a robust controller can be calculated.

5. Measurements and results

5.1. Automatic calibration

The above mentioned method for automatic calibration was implemented and extensively tested using five different types of ball (see Section 2) in order to obtain identified models for each ball. Identification is performed based on a set of step response measurements. The resulting transfer functions for each of the balls are given in Table 2.

Table 2
Results obtained from the automatic calibration.

Ball ID	Weight (g)	K	a	Transfer function
Petanque ball	718	−6.23	1.29	$\frac{-6.23}{s(s+1.29)}$
White-gray pool ball	246	−6.59	0.36	$\frac{-6.59}{s(s+0.36)}$
Hockey ball	152	−6.56	0.93	$\frac{-6.56}{s(s+0.93)}$
Beige pool ball	116	−6.74	0.47	$\frac{-6.74}{s(s+0.47)}$
Ping-pong ball	2.7	−6.41	0.52	$\frac{-6.41}{s(s+0.52)}$

5.2. Robust control

In the beginning of an experiment, this automatic calibration is performed only once. The system automatically detects a change of ball by comparing the radius of the previous ball with the radius of the newly inserted ball. This process is based on image processing techniques such as the Hough transform.

After the automatic calibration, the resulting parameters are used to design the PD controller using the model-based tuning method described in previous section. Once the PD parameters of the controller are optimized they remain the same in order to test the robustness of the system. In the subsequent experiment the automatic calibration is performed on the White-gray pool ball with the resulting transfer function given in Table 2. Based on this transfer function the model-based tuning process optimizes the controller parameters with the following results:

$$\begin{cases} K_p = -0.90 \text{ [rad/m]} \\ K_d = -0.61 \text{ [rad s/m]} \end{cases} \tag{27}$$

Using the same controller parameters for both X–Y axes, the White-gray pool ball is replaced sequentially by all other balls in order to test the robustness of the controller. The reference trajectory is a square form on the plate. The automatic calibration is performed on the White-gray pool ball indicated by '(w/ ID)'. The other trajectories are performed with the controller optimized for the white-gray pool ball while other types of balls are used (indicated by 'w/o ID'). The resulting trajectories are shown in Fig. 6.

The results indicate that the performance of the controller is maintained in presence of significant dynamic variations. Note from Fig. 6 that the rise time and the overshoot is similar for each

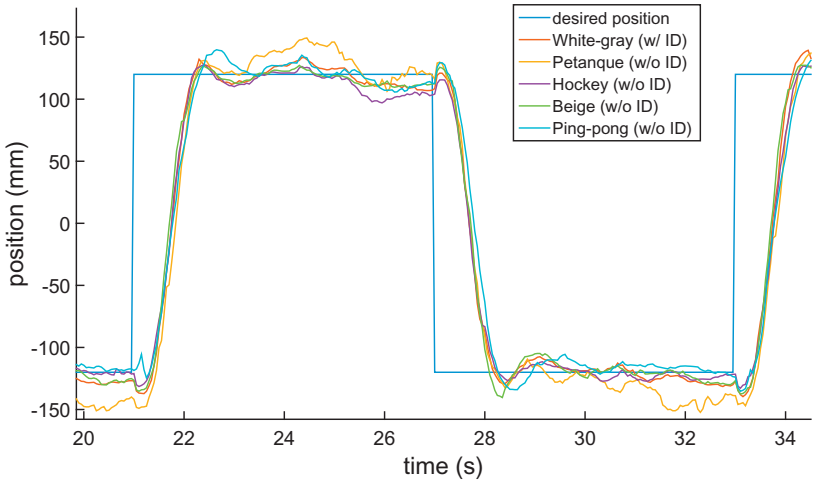


Fig. 6. Output of the system for different balls using a square path as input and the same parameters for the controller. The reference is the position in the *X*–*Y* plane; however, here only position in the *X*-plane is depicted due to the fact that *Y* contains the same information.

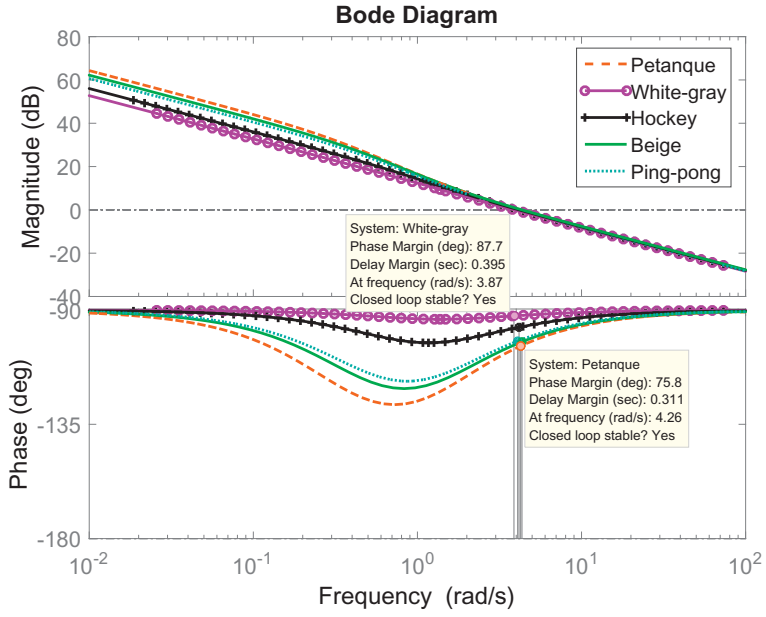


Fig. 7. Bode plots of the open loop systems with the controller tuned for the white-gray pool ball.

of the applied balls. This can be explained by looking at the Bode plot of each open loop system. Fig. 7 shows the Bode plot of each open loop system with the controller tuned based on identification with the white-gray pool ball. Notice from Fig. 7 that the gain cross-over frequencies are the same for each system, explaining why the rise time is similar in Fig. 6 as there exists a direct relation between the gain cross-over frequency and the rise time. Also, from the phase diagram in Fig. 7, it can be noticed that the phase margin is very similar for each system resulting in a similar overshoot which is observed in Fig. 6.

6. Conclusions

This paper presents a robust control approach for an open loop unstable ball and plate system. Using an automatic calibration procedure to extract a model of the process, a model-based tuning technique is used to tune a robust PD controller. The closed loop control robustness is evaluated by a series of tests with different types of balls with distinct differences in weight, diameter and surface. The results indicate that the proposed approach leads to good results irrespective of ball types.

Acknowledgment

C. M. Ionescu acknowledges the Flanders Research Center (FWO) grant no. 12B3415N for its financial support. All authors are members of the Flanders Make consortium, group EEDT.

References

[1] Stewart D. A platform with six degrees of freedom. In: Proceedings of the institution of mechanical engineers; 1965. p. 371–86.
[2] Mura A. Six d.o.f. displacement measuring device based on a modified stewart platform. Mechatronics 2011;21(8):1309–16.
[3] Cheng H, Liu G, Yiu Y, Xiong Z, Li Z. Advantages and dynamics of parallel manipulators with redundant actuation. In: Proceedings of IEEE international conference on intelligent robots and systems; 2001. p. 171–6.
[4] Mianowski K, Wojtyra M. Stewart platform based parallel manipulator destined for the kinematic's justification of two cooperated robots. In: Proceedings of the fourth international workshop on robot motion and control; 2004. p. 115–20.

[5] Shang W, Cong S, Kong F. Identification of dynamic and friction parameters of a parallel manipulator with actuation redundancy. Mechatronics 2010;20(2):192–200.
[6] Wu J, Wang J, Wang L, Li T. Dynamic formulation of redundant and nonredundant parallel manipulators for dynamic parameter identification. Mechatronics 2009;19(4):586–90.
[7] Wu J, Chen X, Li T, Wang L. Optimal design of a 2-DOF parallel manipulator with actuation redundancy considering kinematics and natural frequency. Robot Comput Int Manuf 2013;29(1):80–5.
[8] Wu J, Wang J, Wang L, Li T. Dynamics and control of a planar 3-DOF parallel manipulator with actuation redundancy. Mech Mach Theory 2009;44(4):835–49.
[9] Wu J, Wang D, Wang L. A control strategy of a two degrees-of-freedom heavy duty parallel manipulator. J Dyn Syst Meas Control 2015;137(6):061007.
[10] Grace K, Colgate J, Glucksberg M, Chun J. A six degree of freedom micromanipulator for ophthalmic surgery. In: Proceedings of IEEE international conference on robotics and automation; 1993. p. 630–5.
[11] Wapler M, Urban V, Weisener T, Stallkamp J, Dürr M, Hiller A. A stewart platform for precision surgery. Trans Inst Meas Control 2003;25(4):329–34.
[12] De Paepe M. Design of a ball and plate system with a 6 DoF motion platform and vision-based feedback, [Master's thesis]. Belgium: Ghent University; 2011.
[13] Serway R, Jewett J. Physics for scientists and engineers with modern physics. Cengage Learning; 2013.
[14] Horn A. Doubly stochastic matrices and the diagonal of a rotation matrix. Am J Math 1954;76(3):620–30.
[15] Raptis I, Valavanis K, Moreno W. A novel nonlinear backstepping controller design for helicopters using the rotation matrix. IEEE Trans Control Syst Technol 2011;19(2):465–73.
[16] Rusydi M, Okamoto T, Ito S, Sasaki M. Rotation matrix to operate a robot manipulator for 2d analog tracking objects using electrooculography. Robotics 2014;3(3):298–309.
[17] Rivera D, Gaikwad S. Digital PID controller design using ARX estimation. Comput Chem Eng 1996;20(11):1317–34.
[18] Teunissen P. Dynamic data processing: recursive least-squares. VSSD Publishing; 2009.
[19] Distefano J, Stubberud A, Williams I. Feedback and control systems. McGraw-Hill; 1995.
[20] Toscano R. A simple robust PI/PId controller design via numerical optimization approach. J Process Control 2005;15(1):81–8.