



Tomas Bata University in Zlín
Faculty of Applied Informatics

Doctoral Thesis

Robotické systémy pro rychlé a nestabilní procesy

Robotic Systems in Fast and Unstable Processes

Author: **Ing. Ľuboš Spaček**
Branch of study: Automatic Control and Informatics
Supervisor: doc. Ing. Jiří Vojtěšek, Ph.D.

Zlín, 2022

ABSTRAKT

Práca prezentuje výskum v oblasti robotických systémov riadiacich rýchle a nestabilné procesy s použitím dobre známeho modelu guľôčky na plošine ako referenčného systému. Súčasná robotika sa pomaly posúva smerom alternatívneho použitia robotických manipulátorov v oveľa viac komplexných aplikáciách. Tieto aplikácie vyžadujú väčšiu presnosť sledovania dráhy a rýchlejšiu odozvu než klasické robotické riešenia. Rýchle a nestabilné systémy poskytujú tak ideálny základ pre výskum podobných aplikácií a otvárajú nové možnosti vo využití robotických manipulátorov. Práca prezentuje pilotnú štúdiu uskutočniteľnosti v simulácii a rovnako aj reálne testy na robotickom manipulátore so 7 stupňami voľnosti.

SUMMARY

This thesis presents the research of robotic systems controlling fast and unstable processes using a well-known Ball & Plate model as a reference system. Current robotics is gradually shifting its aim towards alternative methods of using robotic manipulators for more complex applications. These applications require better precision, path accuracy and quicker response time than classic robotic solutions. Fast and unstable systems thus provide an ideal base for research of such applications and open new possibilities in the usage of industrial robots. The thesis presents a pilot and feasibility study in simulation and real tests on the robotic manipulator with 7 degrees of freedom.

TABLE OF CONTENTS

LIST OF FIGURES	7
LIST OF TABLES	9
LIST OF ABBREVIATIONS	10
1 INTRODUCTION	11
1.1 MOTIVATION	11
1.2 STRUCTURE OF THE THESIS	11
2 CURRENT STATE OF THE PROBLEM	12
3 GOALS OF THE THESIS	13
4 THEORETICAL BACKGROUND	14
4.1 MATHEMATICAL MODEL OF BALL & PLATE PROBLEM	15
4.1.1 Setup	15
4.1.2 System Equations	16
4.1.3 Interpretation of System Equations	19
4.1.4 Tensor Form	19
4.1.5 Simplification of the Model	21
4.2 CONTROLLER DESIGN	23
4.2.1 Control Law	25
4.2.2 Optimal Control	27
4.2.3 Spectral Factorization of Polynomials	29
4.3 ROBOT CONTROL	30
5 METHODS	31
5.1 SETUP	31
5.1.1 Robotic Manipulator	31
5.1.2 Sensors	36
5.1.3 B&P Robotic System	37
5.2 ROBOT GUIDED MOTION	40
5.3 VIRTUAL ROBOT SCENE	40
6 EXPERIMENTAL PART	42

6.1	SIMULATED SYSTEM	42
6.1.1	B&P Robotic System Identification	42
6.1.2	Controller Parameters	44
6.1.3	Results	46
6.1.4	Comparison	47
6.1.5	Analysis of Sensitivity to Model Errors	53
6.2	REAL SYSTEM	56
6.2.1	B&P Robotic System Identification	56
6.2.2	Controller Parameters	58
6.2.3	Results	60
7	CONTRIBUTION TO SCIENCE AND PRACTICE	72
8	CONCLUSION	75
	REFERENCES	77
	PUBLICATIONS OF THE AUTHOR	82
	CURRICULUM VITAE.....	84

LIST OF FIGURES

Fig. 4.1	Ball & Plate model setup	15
Fig. 4.2	Structure of the 2 DoF polynomial controller	24
Fig. 5.1	IRB 14000 YuMi Front View Photo	33
Fig. 5.2	IRB 14000 YuMi Front View	33
Fig. 5.3	IRB 14000 YuMi Side View	34
Fig. 5.4	IRB 14000 YuMi Top View	34
Fig. 5.5	IRB 14000 YuMi Isometric View	35
Fig. 5.6	Analog 4-wire resistive touchscreen	37
Fig. 5.7	Load diagram of the B&P setup for robot IRB 14000	38
Fig. 5.8	Robot configuration in a default state	38
Fig. 5.9	B&P setup	39
Fig. 5.10	Dynamics of the plate motion	39
Fig. 5.11	RobotStudio setup of the B&P robotic system	41
Fig. 6.1	Pseudo-identification of the system in simulation	43
Fig. 6.2	Identified parameters for multiple step changes	44
Fig. 6.3	Simulation results for x coordinate	46
Fig. 6.4	Simulation results for y coordinate	47
Fig. 6.5	Position of the ball for a step change	49
Fig. 6.6	Angle of the plate for a step change	49
Fig. 6.7	Position of the ball for a sequence change	50
Fig. 6.8	Angle of the plate for a sequence change	50
Fig. 6.9	Position of the ball for a linear change	51
Fig. 6.10	Angle of the plate for a linear change	51
Fig. 6.11	Position of the ball for a harmonic change	52
Fig. 6.12	Angle of the plate for a harmonic change	52
Fig. 6.13	Sensitivity to change of K parameter	53
Fig. 6.14	Sensitivity to change of Tr parameter	54
Fig. 6.15	Sensitivity to change of both K and Tr parameters	54
Fig. 6.16	Sensitivity to change of transport delay of the controller output	55
Fig. 6.17	Measurements for identification of the system	57
Fig. 6.18	Measurements with the response of averaged coefficients	57

Fig. 6.19	Correlation of identified coefficients for x coordinate	58
Fig. 6.20	Correlation of identified coefficients for y coordinate	58
Fig. 6.21	Control results for x coordinate with motion planner	61
Fig. 6.22	Control results for y coordinate with motion planner	62
Fig. 6.23	Control results for stabilization in x coordinate	63
Fig. 6.24	Control results for stabilization in y coordinate	63
Fig. 6.25	Control results for stabilization in x-y plane	64
Fig. 6.26	Control results for harmonic tracking in x coordinate	65
Fig. 6.27	Control results for harmonic tracking in y coordinate	65
Fig. 6.28	Control results for harmonic tracking in the x-y plane	66
Fig. 6.29	Control results for harmonic tracking in x coordinate	67
Fig. 6.30	Control results for harmonic tracking in y coordinate	67
Fig. 6.31	Control results for harmonic tracking in the x-y plane	68
Fig. 6.32	Control results for harmonic tracking in x coordinate	70
Fig. 6.33	Control results for harmonic tracking in y coordinate	70
Fig. 6.34	Control results for harmonic tracking in the x-y plane	71
Fig. 6.35	Control results for harmonic tracking in the x-y-t space	72

LIST OF TABLES

Tab. 5.1	IRB 14000 - Working range of joints	35
Tab. 5.2	IRB 14000 - Modified D-H parameters	35
Tab. 6.1	Quality of control for a step change	49
Tab. 6.2	Quality of control for a sequence change	50
Tab. 6.3	Quality of control for a linear change	51
Tab. 6.4	Quality of control for a harmonic change	52

LIST OF ABBREVIATIONS

B&P	Ball & Plate
cobot	Collaborative Robot
DoF	Degrees of Freedom
HW	Hardware
IO	Input-Output
LQ	Linear Quadratic
LQR	Linear Quadratic Regulator
MIMO	Multiple-Input, Multiple-Output
PD	Proportional-Derivative (controller)
PID	Proportional-Integral-Derivative (controller)

1 INTRODUCTION

1.1 Motivation

Robotic manipulators are the aim of research efforts in many fields such as industry, military, or medicine. Their main driver is of course industrial automation which requires repetitive tasks handled in the shortest amount of time. They are controlled precisely utilizing their kinematic and dynamic models and even corrections to their mathematical model can be introduced using precise optical measurements to increase their overall absolute accuracy. Most of these manipulators have 6 degrees of freedom achieved by 6 actuators connected in series which allows them to position and reorient their end effectors as needed by the application. Their versatility is probably the main reason for their widespread usage and this thesis examines yet another usage in controlling unstable processes. To precisely control a fast and unstable process can be a very complex task as its behavior can dramatically change according to the particular operating point and initial conditions. A very convenient mechanical representative of an unstable process is Ball & Plate (B&P) system, which is a widely used education-oriented experiment. It is a well-known system mainly used for education, research, and testing because of its scalability and modularity. It combines theory and practice in a straightforward setup and provides insights into various engineering disciplines of mechatronics.

1.2 Structure of the Thesis

The first three chapters of the thesis introduce the problem and provide a survey of the literature for different aspects of the solution. The current state of the problem and the goals of the thesis are presented also. Chapter 4 presents a theoretical and mathematical background of the Ball & Plate model, its interpretation, and subsequent controller design with spectral factorization of polynomials also presented. The chapter also briefly summarizes standard control approaches

for industrial manipulators. Chapter 5 deals with the setup of the virtual scene and the resources needed for the real setup. Chapter 6 shows results from measurements and experiments in both virtual and real environments, identification of parameters of the system, and calculated parameters of the controllers based on algorithms presented in previous chapters. The following chapters conclude the thesis and open discussions of results and their contribution to practice.

2 CURRENT STATE OF THE PROBLEM

There are numerous works on building control strategies for robotic manipulators with standard and general methods ([1],[2],[3],[4]), but newer and more focused control strategies are still being produced ([5],[6]) by a large number of specialists in this field. This topic involves both the academic and private sectors. Numerous studies are also concerned with using robotic manipulators as a black-boxed motion mechanism for robot control relying on external inputs such as force and torque sensors ([7],[8],[9],[10]), optical sensors and cameras ([11],[12]), accelerometers, and other devices ([13],[14]). Besides this, none of the aforementioned works address the response of the robotic manipulator to unstable and relatively fast processes, despite the fact that the need for a robot to control such a process in an industrial setting may arise in the future with rapidly developing technologies such as virtual and augmented reality with tactile feedback for teleoperation of robots ([15],[16],[17],[18],[19]). Remotely-operated industrial robots are also on the rise, particularly in hazardous or remote situations like as offshore oil and gas platforms ([20],[21]). Self-motion of these robots is essential and can involve more delicate jobs supported by more sophisticated algorithms allowing for greater stability and reliability. Despite the fact that these applications do not control unstable processes, it can be assumed that their development will lead to a broader scope of applications that may require such feedback control ([15]) in processes such as polishing, grinding, or deburring in human-robot collaboration tasks or in many advanced applications requiring a non-standard approach to industrial robotic systems [22].

From the traditional 2 degrees of freedom design with actuators coupled in series [23],[24] or parallel [25],[26] configuration, to the 6 degrees of freedom parallel Stewart platform [27],[28], B&P systems can be found in educational, research, and many hobby projects. There are also various options for actuators with a higher degree of freedom, such as [29],[30],[31].

This enormous array of solutions for B&P systems demonstrates that it is a very intriguing and difficult subject to solve, and this dissertation adds to this portfolio of electromechanical structures and their control in order to achieve the objective of ball stabilization and trajectory tracking. Numerous B&P control solutions employ standard PID control or state-space controllers and its variants (PD, LQR) [25],[26],[27],[28]. A double feedback loop structure based on fuzzy logic ([32]), fuzzy supervision and sliding control ([33]), non-linear switching ([34]), and the H-infinity approach ([35]) are examples of "non-standard" solutions.

3 GOALS OF THE THESIS

This thesis provides a preview of industrial robotic systems for controlling fast and unstable processes. A classic Ball & Plate (B&P) model will be constructed and connected to an industrial robotic manipulator as its end effector. A proper control strategy has to be chosen to not only successfully stabilize the ball on the plate and compensate for disturbances, but also to keep the controller effort within certain bounds. B&P models are mostly used at universities or in hobby projects and most of the time only during testing or measurements. However robotic systems are expected to run for long periods and fast and sharp control signals tend to invoke much larger stress on the whole system. For this reason, angular accelerations of generated plate angles (and thus those of joint angles) should be taken into consideration. Proper sensors and control HW should be also chosen.

The following goals are planned to be fulfilled in this thesis:

- To choose a suitable industrial robot for initial simulations, tests, and measurements, which is fast, safe, dexterous in motion, and easy to program and maintain.
- To make a feasibility study and pilot simulations of the robot controlling the B&P model. This should verify whether the chosen robot can perform the given task.
- To choose a suitable B&P system in accordance with the parameters of the robot, which means picking the correct size of the plate and type of the ball. There are many options and possibilities for the setup of B&P, but only one should be considered in this thesis.
- To choose a sensor for obtaining the position of the ball on the plate. This sensor should be fast and reliable enough to follow the dynamics of the B&P model, although the choice depends also on the chosen ball (or vice versa).
- To choose an appropriate control law, that can be easily implemented to the robot's (or controller's) code and is naturally able to manage the control of unstable processes such as the B&P model.
- To achieve a satisfying trajectory-tracking and path-following.

4 THEORETICAL BACKGROUND

Theoretical knowledge in this chapter covers the mathematical model of the B&P system, its simplification and linearization, the design of the controller, and a brief introduction to the control of robotic manipulators with literature references for further reading.

4.1 Mathematical Model of Ball & Plate Problem

Ball & Plate mathematical model is derived from the proposed setup in this chapter. The aim is to obtain a model suitable for control within the given criteria which will be used in the following chapters.

4.1.1 Setup

The B&P problem can be separated into multiple logical segments, each describing a specific part of the B&P model. This division best serves for the derivation of system equations. The best approach in the case of the B&P model is to divide it into two separate directional components x and y . The model setup of such a system for one coordinate is depicted in Fig. 4.1 which describes the motion of a ball on a plate in one of these dimensions. The position of the ball $(x, y [m])$ is expressed in the local coordinate system of the plate $(x - z)$, defined in the right-handed Cartesian base frame $x_0 - z_0$. The plate can rotate around the center of this local coordinate system by the angle α (and β for a 2-dimensional system). The ball, described by its radius $r [m]$ and velocity $\dot{x} [ms^{-1}]$ moves on the plate with gravitational acceleration $g [ms^{-2}]$ acting on the ball.

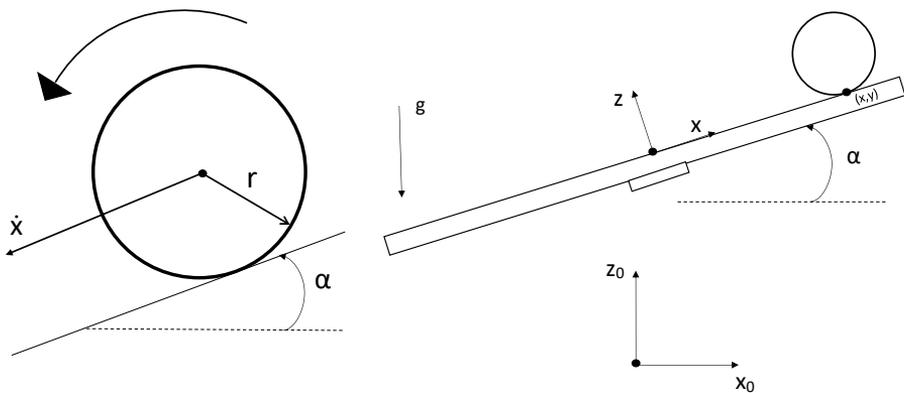


Fig. 4.1 Ball & Plate model setup

Several aspects of the described setup are not mentioned as they are neglected in the favor of simplifying the model. All these assumptions eliminate negligible forces in comparison to gravitational or centrifugal ones and provide a substantial simplification of the problem. This simplification helps with the overall design and implementation of the controller and neglected effects on the ball (that could change its dynamics), are thus considered disturbances in the system. The following derivation of the mathematical model considers several assumptions to simplify the description:

- The air friction is neglected.
- The friction between the ball and the plate is neglected.
- The ball is a homogeneous, ideal sphere (or spherical shell).
- The plate has no boundaries and stretches infinitely long.
- There is always a slip-less contact between the ball and the plate.

4.1.2 System Equations

The general form of the Euler-Lagrange equation of the second kind shown in (4.1) is used to obtain equations of the B&P model:

$$\frac{d}{dt} \frac{\partial T}{\partial \dot{q}_i} - \frac{\partial T}{\partial q_i} + \frac{\partial V}{\partial q_i} = Q_i \quad (4.1)$$

where T is the overall kinetic energy of the system, V is potential energy, Q_i is a generalized force, and q_i is a generalized coordinate. B&P system has four generalized coordinates in total - two position coordinates [$x = q_x, y = q_y$] and two plate inclinations [$\alpha = q_\alpha, \beta = q_\beta$], in which the center of the plate has coordinates [x, y] = $[0, 0]$. The only external force influencing the ball (taking into account assumptions from chapter 4.1.1) is a gravitational force, thus forces [Q_x, Q_y] = $[0, 0]$. The system also experiences torques on the plate [$\tau_\alpha = Q_\alpha, \tau_\beta = Q_\beta$].

Kinetic energies of the ball and plate make up the overall kinetic energy of the system, as shown in (4.2):

$$T = T_{ball} + T_{plate} \quad (4.2)$$

The kinetic energy of the ball has translational and rotational components (see 4.3), which are presented in equations (4.4) and (4.5) respectively:

$$T_{ball} = T_{trans} + T_{rot} \quad (4.3)$$

$$T_{trans} = \frac{1}{2}mv^2 = \frac{1}{2}m(\dot{x}^2 + \dot{y}^2) \quad (4.4)$$

$$T_{rot} = \frac{1}{2}I_b\omega^2 = \frac{1}{2}I_b\frac{v^2}{r^2} = \frac{1}{2}\frac{I_b}{r^2}(\dot{x}^2 + \dot{y}^2) \quad (4.5)$$

where v and ω are linear and angular velocities of the ball respectively, I_b is its moment of inertia, r is the radius, and m is the mass of the ball. The combination of equations (4.3) - (4.5) leads to (4.6):

$$T_{ball} = \frac{1}{2}m(\dot{x}^2 + \dot{y}^2) + \frac{1}{2}\frac{I_b}{r^2}(\dot{x}^2 + \dot{y}^2) = \frac{1}{2}(m + \frac{I_b}{r^2})(\dot{x}^2 + \dot{y}^2) \quad (4.6)$$

The kinetic energy of the plate is dependent on the position of the ball on the plate and its own moment of inertia I_p , as shown in (4.7):

$$T_{plate} = \frac{1}{2}(I_b + I_p)(\dot{\alpha}^2 + \dot{\beta}^2) + \frac{1}{2}m(\dot{\alpha}x + \dot{\beta}y)^2 \quad (4.7)$$

By substituting (4.6) and (4.7) into (4.2) the overall kinetic energy of the system is obtained (see (4.8)):

$$T = \frac{1}{2}(m + \frac{I_b}{r^2})(\dot{x}^2 + \dot{y}^2) + \frac{1}{2}(I_b + I_p)(\dot{\alpha}^2 + \dot{\beta}^2) + \frac{1}{2}m(\dot{\alpha}x + \dot{\beta}y)^2 \quad (4.8)$$

The potential energy of the system is shown in (4.9):

$$V = mgh = mg(x \sin \alpha + y \sin \beta) \quad (4.9)$$

Individual components of the Euler-Lagrange equation (4.1) are shown in equations (4.10) - (4.16)

$$\frac{\partial T}{\partial \dot{x}} = (m + \frac{I_b}{r^2})\dot{x}, \quad \frac{\partial T}{\partial \dot{y}} = (m + \frac{I_b}{r^2})\dot{y} \quad (4.10)$$

$$\frac{\partial T}{\partial \dot{\alpha}} = m(\dot{\alpha}x^2 + \dot{\beta}xy) + (I_b + I_p)\dot{\alpha}, \quad \frac{\partial T}{\partial \dot{\beta}} = m(\dot{\alpha}xy + \dot{\beta}y^2) + (I_b + I_p)\dot{\beta} \quad (4.11)$$

$$\frac{d}{dt} \frac{\partial T}{\partial \dot{x}} = (m + \frac{I_b}{r^2})\ddot{x}, \quad \frac{d}{dt} \frac{\partial T}{\partial \dot{y}} = (m + \frac{I_b}{r^2})\ddot{y} \quad (4.12)$$

$$\frac{d}{dt} \frac{\partial T}{\partial \dot{\alpha}} = (I_b + I_p + mx^2)\ddot{\alpha} + m(\ddot{\beta}xy + \dot{\beta}(\dot{x}y + x\dot{y}) + 2\dot{\alpha}\dot{x}) \quad (4.13)$$

$$\frac{d}{dt} \frac{\partial T}{\partial \dot{\beta}} = (I_b + I_p + my^2)\ddot{\beta} + m(\ddot{\alpha}xy + \dot{\alpha}(\dot{x}y + x\dot{y}) + 2\dot{\beta}\dot{y}) \quad (4.14)$$

$$\frac{\partial T}{\partial x} = m(\dot{\alpha}\dot{\beta}y + \dot{\alpha}^2x), \quad \frac{\partial T}{\partial y} = m(\dot{\alpha}\dot{\beta}x + \dot{\beta}^2y), \quad \frac{\partial T}{\partial \alpha} = 0, \quad \frac{\partial T}{\partial \beta} = 0 \quad (4.15)$$

$$\frac{\partial V}{\partial x} = mg \sin \alpha, \quad \frac{\partial V}{\partial y} = mg \sin \beta, \quad \frac{\partial V}{\partial \alpha} = mgx \cos \alpha, \quad \frac{\partial V}{\partial \beta} = mgy \cos \beta \quad (4.16)$$

System equations for all four generalized coordinates are depicted in equations (4.17) - (4.20):

$$\mathbf{x} : (m + \frac{I_b}{r^2})\ddot{x} - m(\dot{\alpha}\dot{\beta}y + \dot{\alpha}^2x) + mg \sin \alpha = 0 \quad (4.17)$$

$$\mathbf{y} : (m + \frac{I_b}{r^2})\ddot{y} - m(\dot{\alpha}\dot{\beta}x + \dot{\beta}^2y) + mg \sin \beta = 0 \quad (4.18)$$

$$\mathbf{\alpha} : (I_b + I_p + mx^2)\ddot{\alpha} + m(\ddot{\beta}xy + \dot{\beta}(\dot{x}y + x\dot{y}) + 2\dot{\alpha}\dot{x}) + mgx \cos \alpha = \tau_\alpha \quad (4.19)$$

$$\mathbf{\beta} : (I_b + I_p + my^2)\ddot{\beta} + m(\ddot{\alpha}xy + \dot{\alpha}(\dot{x}y + x\dot{y}) + 2\dot{\beta}\dot{y}) + mgy \cos \beta = \tau_\beta \quad (4.20)$$

These nonlinear system equations describe the motion of the ball on the plate in relation to the angle of the plate and its angular velocity (equations (4.17) and (4.18)). On the other hand, equations (4.19) and (4.20) show the dynamics of the plate and its dependency on external torques and the ball's position.

4.1.3 Interpretation of System Equations

A summary of used symbols, their units, and physical analogies of components in equations is presented below:

- m Mass of the ball [kg]
- r Radius of the ball [m]
- g Gravitational acceleration [$m.s^{-2}$]
- I_b, I_p Moments of inertia of the ball and plate [$kg.m^2$]
- x, y Position of the ball relative to the plate center [m]
- \dot{x}, \dot{y} First time derivatives of coordinates [$m.s^{-1}$]
- \ddot{x}, \ddot{y} Second time derivatives of coordinates [$m.s^{-2}$]
- α, β Plate angles [rad]
- $\dot{\alpha}, \dot{\beta}$ First time derivatives of plate angles [$rad.s^{-1}$]
- $\ddot{\alpha}, \ddot{\beta}$ Second time derivatives of plate angles [$rad.s^{-2}$]
- τ_α, τ_β Plate torques [$N.m$]
- $m(\dot{\alpha}\dot{\beta}y + \dot{\alpha}^2x)$ Centrifugal force [N]
- $(I_b + I_p + mx^2)\ddot{\alpha}$ Combined inertia torque [$N.m$]
- $m(\ddot{\beta}xy + \dot{\beta}(\dot{x}y + x\dot{y}))$ Gyroscopic influence [$N.m$]
- $2m\dot{\alpha}\dot{x}$ Coriolis influence [$N.m$]
- $mgx \cos \alpha$ Gravitational influence [$N.m$]

4.1.4 Tensor Form

Torques acting on the B&P system have four different origins:

- Dynamic - inertial, centrifugal, Coriolis
- Static - not present in the model (caused by friction)
- Gravitational
- External - mainly from the actuators of the system

This categorization can be used to write an alternative description to system equations in tensor form, shown in equation (4.21):

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = Q \quad (4.21)$$

where $M(q)$ is an inertia matrix, $C(q, \dot{q})$ stands for Coriolis matrix (consisting of Coriolis and centrifugal forces), $G(q)$ is a gravity matrix and Q is a vector of external forces. Individual matrices are shown in equations (4.22) - (4.25):

$$q = \begin{bmatrix} x \\ y \\ \alpha \\ \beta \end{bmatrix}, \quad \dot{q} = \begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\alpha} \\ \dot{\beta} \end{bmatrix}, \quad \ddot{q} = \begin{bmatrix} \ddot{x} \\ \ddot{y} \\ \ddot{\alpha} \\ \ddot{\beta} \end{bmatrix}, \quad \dot{Q} = \begin{bmatrix} 0 \\ 0 \\ \tau_\alpha \\ \tau_\beta \end{bmatrix} \quad (4.22)$$

$$M(q) = \begin{bmatrix} \left(m + \frac{I_b}{r^2}\right) & 0 & 0 & 0 \\ 0 & \left(m + \frac{I_b}{r^2}\right) & 0 & 0 \\ 0 & 0 & (I_p + I_b + mx^2) & mxy \\ 0 & 0 & mxy & (I_p + I_b + my^2) \end{bmatrix} \quad (4.23)$$

$$C(q, \dot{q}) = m \begin{bmatrix} 0 & 0 & -\dot{\alpha}x & -\dot{\alpha}y \\ 0 & 0 & -\dot{\beta}x & -\dot{\beta}y \\ 2\dot{\alpha}x & 0 & 0 & (\dot{x}y + x\dot{y}) \\ 0 & 2\dot{\beta}y & (\dot{x}y + x\dot{y}) & 0 \end{bmatrix} \quad (4.24)$$

$$G(q) = mg \begin{bmatrix} \sin \alpha \\ \sin \beta \\ x \cos \alpha \\ x \cos \beta \end{bmatrix} \quad (4.25)$$

4.1.5 Simplification of the Model

Assumptions from chapter 4.1.1 need to be taken into account to simplify the model for the purposes of controller design. The model can be linearized around the point where plate angles are near zero and the rate of change of these angles is also nearing zero. In such cases it is possible to assume:

- $|\alpha| \ll 1 \Rightarrow \sin \alpha \approx \alpha$
- $|\beta| \ll 1 \Rightarrow \sin \beta \approx \beta$
- $|\dot{\alpha}| \ll 1; |\dot{\beta}| \ll 1 \Rightarrow \dot{\alpha}\dot{\beta} \approx 0; \dot{\alpha}^2 \approx 0; \dot{\beta}^2 \approx 0$

These assumptions remove the effects of centrifugal forces and most importantly remove the dependence of x and y coordinates on each other, which leads to easy separation of these coordinates in differential equations, making the result symmetric in positional coordinates. Equations (4.26) and (4.27) originated by substituting the assumptions into equations (4.17) and (4.18):

$$\mathbf{x} : \left(m + \frac{I_b}{r^2}\right)\ddot{x} + mg\alpha = 0 \quad (4.26)$$

$$\mathbf{y} : \left(m + \frac{I_b}{r^2}\right)\ddot{y} + mg\beta = 0 \quad (4.27)$$

It is trivial to get the moment of inertia of the ball, by assuming it is a perfect homogeneous sphere (or spherical shell). Moments of inertia for the full sphere

and spherical shell can be seen in (4.28) and (4.29):

$$I_{sphere} = \frac{2}{5}mr^2 \quad (4.28)$$

$$I_{shell} = \frac{2}{3}mr^2 \quad (4.29)$$

It is possible to see the resulting model (equations (4.30) and (4.31)) is not dependent on the mass or dimensions of the ball by substituting (4.28) into (4.26) and (4.27):

$$\mathbf{x} : \frac{7}{5}\ddot{x} + g\alpha = 0 \quad (4.30)$$

$$\mathbf{y} : \frac{7}{5}\ddot{y} + g\beta = 0 \quad (4.31)$$

Equations (4.32) and (4.33) are the result of the reorganization of the equation and substitution of constants by a single constant K_b :

$$\mathbf{x} : \ddot{x} = K_b\alpha \quad (4.32)$$

$$\mathbf{y} : \ddot{y} = K_b\beta \quad (4.33)$$

where constant K_b for a full spherical homogeneous ball is shown in (4.34):

$$K_b = -\frac{5}{7}g \quad (4.34)$$

where g is a gravitational acceleration constant. Equations (4.32) and (4.33) are linear differential equations and can be easily expressed in Laplace format as shown in (4.35) and (4.36) and a state-space form in (4.37):

$$\mathbf{x} : G_x(s) = \frac{K_b}{s^2} \quad (4.35)$$

$$\mathbf{y} : G_y(s) = \frac{K_b}{s^2} \quad (4.36)$$

$$\begin{aligned} \begin{bmatrix} \dot{x} \\ \ddot{x} \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \end{bmatrix} + \begin{bmatrix} 0 \\ K_b \end{bmatrix} \alpha \\ y &= \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \end{bmatrix} \end{aligned} \quad (4.37)$$

As for the second part of the system equations ((4.19) and (4.20)), which describe the dynamics of the plate itself and the effects of the ball on the plate, they are omitted in this description of the B&P mathematical model, as compensation of these effects is handled by the drives of the robotic system. They are instead replaced by an approximation in the form of a first-order transfer function (4.38) and merged with equations (4.35) and (4.36) to create a general model of the B&P system (seen in (4.39)), which can be used for identification and controller synthesis in following chapters, where subscript letter r stands for robot and b for the ball. A discrete version of this model is also presented in (4.40) with unknown coefficients b_i and a_i .

$$G_r(s) = \frac{K_r}{T_r s + 1} \quad (4.38)$$

$$G(s) = \frac{K_b}{s^2} \frac{K_r}{T_r s + 1} = \frac{K}{s^2(T_r s + 1)} = \frac{K}{T_r s^3 + s^2} \quad (4.39)$$

$$G(z^{-1}) = \frac{B(z^{-1})}{A(z^{-1})} = \frac{b_1 z^{-1} + b_2 z^{-2} + b_3 z^{-3}}{1 + a_1 z^{-1} + a_2 z^{-2} + a_3 z^{-3}} \quad (4.40)$$

4.2 Controller Design

A linear-quadratic (LQ) controller with a polynomial 2 degree of freedom (DoF) structure is the main controller design method used in this thesis. It was chosen as the solution for this type of problem, based mainly on a comparison of different controllers in previous works of the author [P.7]. Important criteria for

the controller, in this case, are robustness, good disturbance rejection, and low controller effort (rate of change of its output). Low controller effort is probably the most important parameter because repeated and frequent usage is considered for the robotic system and low impacts on its mechanics (transmissions mostly) should be kept in mind during the design, thus abrupt angle changes should be avoided if possible. The structure of the 2 DoF controller contains a feed-forward filter which helps to reduce the impacts of sudden changes in reference value. Fig. 4.2 shows the structure of a closed-loop polynomial 2 DoF discrete-time controller, where G is the linearized B&P model itself (4.40), C_f is the feedforward component of the controller, responsible for reference tracking, C_b is the feedback component of the controller, responsible for stabilization and disturbance rejection, $\frac{1}{(1-z^{-1})}$ is a summation part extracted out of C_f and C_b components (so the summation happens after the subtraction of these components), $w(k)$ is a reference value (desired position of the ball), $u(k)$ is a controller output (plate angle), $y(k)$ is output value (position of the ball) and $n(k)$ and $v(k)$ are disturbances acting on the system. These disturbances are the result of random external forces that are not part of the model and measurement errors in the position of the ball. They also envelop simplifications of the model whole model, namely $n(k)$ contains errors in the expected model of the actuator system (robot in this case) and $v(k)$ accounts for errors in the identification of the B&P system and its approximation.

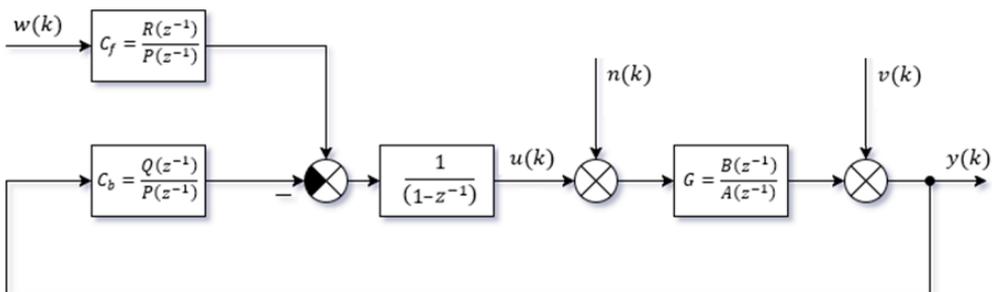


Fig. 4.2 Structure of the 2 DoF polynomial controller

4.2.1 Control Law

Equations (4.35) and (4.36) show the linearized system is symmetric, thus the control law definition is described for a one-dimensional case which will be later applied to both coordinates. Equations (4.41) expressing the output of the system (position of the ball) and (4.42) expressing the output of the controller (plate angle) are used for the subsequent description of the controller shown in Fig. 4.2:

$$Y(z^{-1}) = \frac{B(z^{-1})}{A(z^{-1})}U(z^{-1}) \quad (4.41)$$

$$U(z^{-1}) = \frac{R(z^{-1})}{(1 - z^{-1})P(z^{-1})}W(z^{-1}) - \frac{Q(z^{-1})}{(1 - z^{-1})P(z^{-1})}Y(z^{-1}) \quad (4.42)$$

The structure of the controller can be thus expressed in the polynomial equation (4.43) relating the position of the ball to the reference value and the polynomial equation (4.44) relating the plate angle to the reference value:

$$Y(z^{-1}) = \frac{B(z^{-1})R(z^{-1})}{A(z^{-1})(1 - z^{-1})P(z^{-1}) + B(z^{-1})Q(z^{-1})}W(z^{-1}) \quad (4.43)$$

$$U(z^{-1}) = \frac{A(z^{-1})R(z^{-1})}{A(z^{-1})(1 - z^{-1})P(z^{-1}) + B(z^{-1})Q(z^{-1})}W(z^{-1}) \quad (4.44)$$

Equation (4.43) contains a characteristic polynomial in its denominator $D(z^{-1})$ shown in (4.45) and the pole assignment algebraic method of controller design is followed as described in [36]:

$$D(z^{-1}) = A(z^{-1})(1 - z^{-1})P(z^{-1}) + B(z^{-1})Q(z^{-1}) \quad (4.45)$$

It is important to know the degrees of polynomials $A(z^{-1})$ and $B(z^{-1})$ to determine the degree of polynomials $Q(z^{-1})$ and $P(z^{-1})$. The degree of the characteristic polynomial $D(z^{-1})$ in (4.45) can be chosen, but it is recommended

in [36] to meet at least the minimum criteria of (4.46):

$$\partial D(z^{-1}) \leq \partial A(z^{-1}) + \partial B(z^{-1}) \quad (4.46)$$

The degree of polynomials $A(z^{-1})$ and $B(z^{-1})$ is determined from (4.40) and holds:

$$\partial A(z^{-1}) = 3; \quad \partial B(z^{-1}) = 3 \quad (4.47)$$

This leads to a determination of the degree of the characteristic polynomial from (4.46) to be at least 6. The structure of the characteristic polynomial $D(z^{-1})$ is thus:

$$D(z^{-1}) = \sum_{i=0}^6 d_i z^{-i} = d_0 + d_1 z^{-1} + d_2 z^{-2} + d_3 z^{-3} + d_4 z^{-4} + d_5 z^{-5} + d_6 z^{-6} \quad (4.48)$$

Degrees of polynomials ∂Qz^{-1} and ∂Pz^{-1} are determined by inserting the degree of $\partial Dz^{-1} = 6$ and degrees of Az^{-1} and Bz^{-1} (4.47) into equation (4.45):

$$\partial D(z^{-1}) = 6; \quad \partial Q(z^{-1}) = 3; \quad \partial P(z^{-1}) = 2 \quad (4.49)$$

The only remaining unknown is the degree of $R(z^{-1})$, which is according to [36] dependent on the denominator of the reference signal (4.50). Two reference signals are considered - step change (4.51) and harmonic signal (4.52):

$$\partial R(z^{-1}) = \partial W_{denom} - 1 \quad (4.50)$$

$$W_{step}(z^{-1}) = \frac{1}{1 - z^{-1}} \Rightarrow \partial R(z^{-1}) = 0 \quad (4.51)$$

$$W_{harm}(z^{-1}) = \frac{z^{-1} \sin \omega_0}{1 - 2z^{-1} \cos \omega_0 + z^{-2}} \Rightarrow \partial R(z^{-1}) = 1 \quad (4.52)$$

Feedforward and feedback components of the controller are thus (4.53) and (4.54), where the denominator $P(z^{-1})$ is expressed in a form that has coefficient $p_0 = 1$ (based on previous experience and practical simplification of

further calculations). Expression (4.42) can be thus written as (4.55) for the sake of implementation to the code.

$$C_f(z^{-1}) = \frac{R(z^{-1})}{P(z^{-1})} = \frac{r_0}{1 + p_1z^{-1} + p_2z^{-2}} \quad (4.53)$$

$$C_b(z^{-1}) = \frac{Q(z^{-1})}{P(z^{-1})} = \frac{q_0 + q_1z^{-1} + q_2z^{-2} + q_3z^{-3}}{1 + p_1z^{-1} + p_2z^{-2}} \quad (4.54)$$

$$\begin{aligned} u(k) = & r_0w(k) - q_0y(k) - q_1y(k-1) - q_2y(k-2) - q_3y(k-3) + \\ & + (1 - p_1)u(k-1) + (p_1 - p_2)u(k-2) + p_2u(k-3) \end{aligned} \quad (4.55)$$

4.2.2 Optimal Control

Coefficients of the characteristic polynomial (4.48) would be placed by the user accordingly, using the pole-placement method, but to achieve a more optimal solution, one can use the linear quadratic method which tries to minimize the criterion (4.56) while penalizing the controller's output $u(k)$.

$$J = \sum_{k=0}^{\infty} \left\{ [e(k)]^2 + q_u [u(k)]^2 \right\} \quad (4.56)$$

where $e(k) = w(k) - y(k)$ is the control error, $u(k)$ is controller output and q_u is a penalization constant. Minimization of this criterion leads to the solution of the Riccati equation, but another approach can be exploited using spectral factorization of polynomials by solving two matrix polynomial equations [37]. The solution is based on polynomial input-output models and is very suitable for systems with unknown or inaccessible states. According to [37], when the error and controller output signals are expanded into polynomials, the criterion (4.56) can be rewritten to (4.57):

$$J = \langle E(z)E(z^{-1}) + q_uU(z)U(z^{-1}) \rangle \quad (4.57)$$

where $E(z)$ and $U(z)$ are polynomials with respective negative powers replaced by positive ones. Expressions (4.43) and (4.44) can be then substituted into this criterion and it is proven in [37] that such a criterion is minimized for (4.45), where $D(z^{-1})$ comes from the spectral factorization of equation (4.58).

$$A(z^{-1})q_u A(z) + B(z^{-1})B(z) = D(z^{-1})\delta D(z) \quad (4.58)$$

where δ is a constant added to obtain $D(z^{-1})$ with $d_0 = 1$. This solution obtains only half of the optimal coefficients of $D(z^{-1})$ in most cases, as the degree of solution of (4.58) is bound by degrees of $A(z^{-1})$ or $B(z^{-1})$ polynomials. Thus only three optimal poles are obtained by this method for the B&P model (4.40) and others have to be placed. Poles placed closer to the unit circle cause more subtle changes in plate angles and vice versa.

The Diophantine equation (4.45) can be thus solved by comparing coefficients of individual z^{-i} powers on both sides of the equation. This creates a system of linear equations (4.59) - 6 in total in the case of (4.48), while $d_0 = 1$ due to the influence of δ in (4.58).

$$\begin{bmatrix} b_3 & 0 & 0 & 0 & -a_3 & 0 \\ b_2 & b_3 & 0 & 0 & a_3 - a_2 & -a_3 \\ b_1 & b_2 & b_3 & 0 & a_2 - a_1 & a_3 - a_2 \\ 0 & b_1 & b_2 & b_3 & a_1 - 1 & a_2 - a_1 \\ 0 & 0 & b_1 & b_2 & 1 & a_1 - 1 \\ 0 & 0 & 0 & b_1 & 0 & 1 \end{bmatrix} \begin{bmatrix} q_3 \\ q_2 \\ q_1 \\ q_0 \\ p_2 \\ p_1 \end{bmatrix} = \begin{bmatrix} d_6 \\ d_5 \\ d_4 + a_3 \\ d_3 - a_3 + a_2 \\ d_2 - a_2 + a_1 \\ d_1 - a_1 + 1 \end{bmatrix} \quad (4.59)$$

The coefficient r_0 of $R(z^{-1})$ can be calculated for step-changing value from other coefficients of the Diophantine equation, according to [36], as seen in (4.60):

$$r_0 = \frac{\sum_{i=0}^6 d_i}{\sum_{i=1}^3 b_i} \quad (4.60)$$

Coefficients can be then substituted into equations (4.53), (4.54), and (4.55).

4.2.3 Spectral Factorization of Polynomials

The spectral factorization leaves the stable part of the polynomial unchanged and changes only the unstable one. It is possible to calculate spectral factorization analytically for a maximum of second-degree polynomials and iterative methods need to be used for higher-order polynomials [36]. By replacing the known left part of the equation (4.58) with a single component, its resulting form is shown in (4.61):

$$M(z^{-1})M(z) = D(z^{-1})\delta D(z) \quad (4.61)$$

where $M(z)$ and $D(z)$ are conjugate pairs of their counterparts, meaning they have negative powers of z replaced with positive ones. For first-order polynomials, the equation would be:

$$(m_0 + m_1z^{-1})(m_0 + m_1z) = (1 + d_1z^{-1})\delta(1 + d_1z) \quad (4.62)$$

$$(m_0^2 + m_1^2) + m_0m_1(z + z^{-1}) = \delta(1 + d_1^2) + \delta d_1(z + z^{-1}) \quad (4.63)$$

A simple system of equations is obtained by comparing coefficients of $(z + z^{-1})$ terms, as seen in (4.64). Coefficients d_1 and δ can be subsequently calculated.

$$\begin{aligned} m_0^2 + m_1^2 &= \delta(1 + d_1^2) \\ m_0m_1 &= \delta d_1 \end{aligned} \quad (4.64)$$

This derivation serves only as an example to understand the nature of spectral factorization of polynomials because the B&P description contains polynomials of 3rd degree (see (4.40)) and their spectral factorization is calculated numerically. This task can be achieved by Polynomial Toolbox in MATLAB [38] which is a package for system analysis based on advanced polynomial methods. It consists of many tools, functions, and built-in design routines:

- Polynomial matrix operations.
- Pre-defined variables such as s and z .

- Polynomial matrix editor for large matrices.
- Matrix polynomial equation solvers.
- Spectral factorization algorithms.
- Diophantine and Riccati equation solvers.
- Polynomial Matrix Fractions support.
- Analysis tools (robustness, stability margins, parametric and polytopic uncertainties).

4.3 Robot control

Although the robot control itself is not directly part of this thesis, the author feels important to introduce at least basic concepts of controlling individual joints of a robotic manipulator to achieve desired position control. This brief introduction is supported by [3] and [4] which contain a thorough explanation of manipulator stability, dynamic modeling, and control strategies for different kinematic structures and multiple examples of manipulators with differing degrees of freedom. The dynamic description of the robotic manipulator is mainly based on Lagrange equations, such as the one mentioned in chapter 4.1.2, and with dynamic model description in tensor form, as shown in chapter 4.1.4, consisting of inertia matrix, centrifugal and Coriolis forces matrix, gravitational torques vector and residual dynamics.

Classical position control of manipulators is dependent on the kinematic and dynamic model of the manipulator but can be in most cases achieved by some sort of PD control. Specifically [4] distinguishes position (set-point) control and motion (path tracking) control:

- Proportional control with velocity feedback
- PD control

- PD control with gravity compensation
- PD control with desired gravity compensation
- PID control
- Feedforward + PD control for path tracking

Adaptive versions of these controllers are required for more advanced scenarios without velocity measurements or those with model uncertainty [4].

5 METHODS

Methods used in the setup of the experimental part of the thesis are described in this chapter. It starts with the setup of the whole B&P system and describes SW tools to guide the robot and simulate the application.

5.1 Setup

This subsection describes the hardware setup of the B&P robotic manipulator system used in this thesis.

5.1.1 Robotic Manipulator

The solution to the B&P problem requires an extensive testing phase on the real system and rapid robot movements with a fast acceleration of multiple robot joints in several directions (B&P is a relatively fast and more importantly unstable process) may pose a threat to the tester (the author) without safety countermeasures. However, these countermeasures may limit the testing process (mainly the access to the robot itself) and thus a safer solution is required than standard industrial robotic manipulators. So-called collaborative robots

are emerging in recent years as standard industrial solutions for collaborative or cooperative applications requiring no fences and introducing a new type of industrial robot to the market, with torque supervision capabilities - cobots [39][40][41][42].

Collaborative robot ABB IRB 14000 YuMi is used for testing in this thesis. It is a relatively small, dual-arm industrial robot with a handling capacity of 0.5 kg and 0.559 m reach. Each arm has 7 degrees of freedom and because of its built-in torque sensors is relatively safe for operation even at higher speeds. Figs. 5.1-5.5 show the robot and its 2D views with dimensions of maximum and minimum reach listed in *mm* [43][44]. The majority of industrial robots have 6-DoF configurations with motors connected in series, which meets most industrial requirements. Several cobots use the 7th axis to improve the reachability and flexibility of movement of the manipulator. Control of the B&P model using 7 degrees of freedom robotic manipulator (Robai Cyton Gamma 300) is tested in [29], but only the last 2 axes are used for control and others are stationary. This thesis aims to use the full extent of all 7 axes and although several of them might be moving only slightly they are still being used in kinematic calculations and their dynamics influence the whole system. In addition, they can be used to move the whole plate in space while balancing the ball.

The range of motion of its individual joints is presented in Tab. 5.1 [43] and kinematic model of one arm expressed in Modified Denavit-Hartenberg notation in Tab. 5.2 [45], where q_i is a joint variable (angle in this case). Dynamic parameters (inertial and friction parameters) of the manipulator are identified and thoroughly described in [45] for further reading.

The programming language of the robot is called RAPID, which is a scripting language and provides basic functionalities for controlling the movements of the robot, signal handling, communication over various interfaces, conditional statements, loops, structures, and other essential procedures for the execution of standard or advanced motion routines.



Fig. 5.1 IRB 14000 YuMi Front View Photo

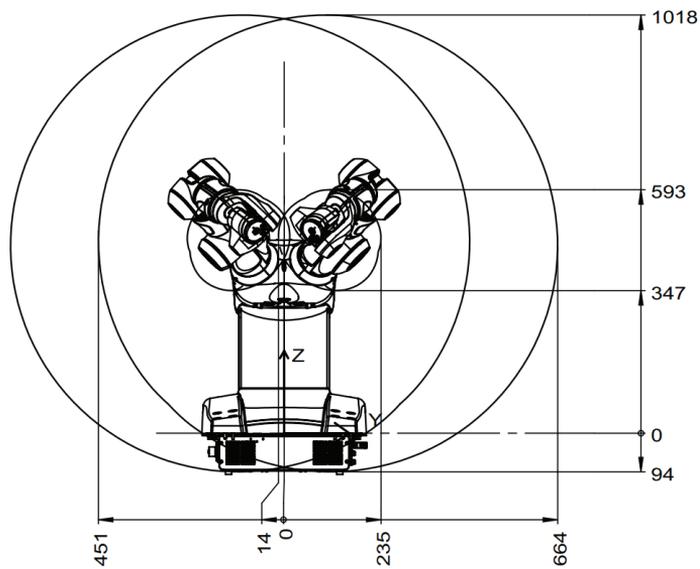


Fig. 5.2 IRB 14000 YuMi Front View

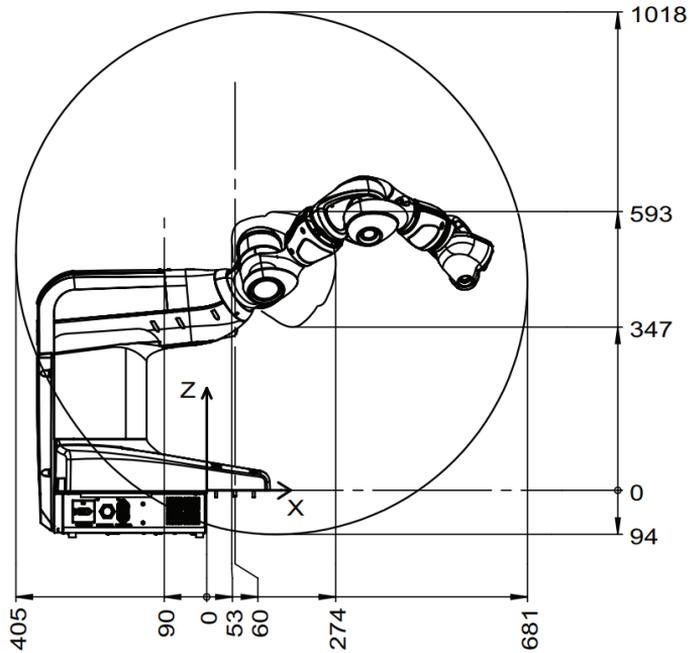


Fig. 5.3 IRB 14000 YuMi Side View

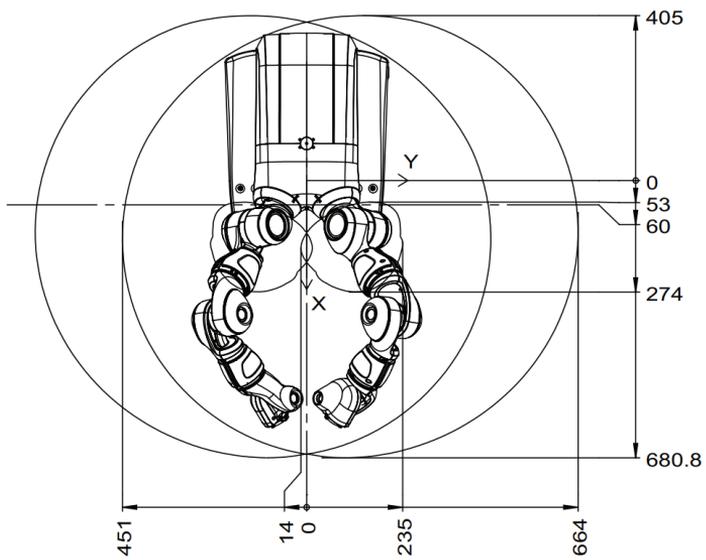


Fig. 5.4 IRB 14000 YuMi Top View

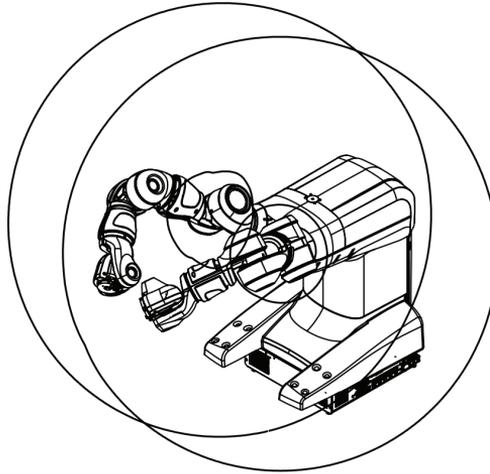


Fig. 5.5 IRB 14000 YuMi Isometric View

Tab. 5.1 IRB 14000 - Working range of joints

Axis 1	Arm - Rotation motion	-168.5° to $+168.5^\circ$
Axis 2	Arm - Bend motion	-143.5° to $+43.5^\circ$
Axis 7	Arm - Rotation motion	-168.5° to $+168.5^\circ$
Axis 3	Arm - Bend motion	-123.5° to $+80^\circ$
Axis 4	Wrist - Rotation motion	-290° to $+290^\circ$
Axis 5	Wrist - Bend motion	-88° to $+138^\circ$
Axis 6	Flange - Rotation motion	-229° to $+229^\circ$

Tab. 5.2 IRB 14000 - Modified D-H parameters

Link	$\alpha_i [rad]$	$d_i [cm]$	$r_i [cm]$	$\theta_i [rad]$
1	0	0	16.6	$q_1 - \pi$
2	$\frac{\pi}{2}$	3	0	$q_2 - \pi$
3	$\frac{\pi}{2}$	3	25.15	q_3
4	$-\frac{\pi}{2}$	4.05	0	$q_4 - \frac{\pi}{2}$
5	$-\frac{\pi}{2}$	4.05	26.5	$q_5 + \pi$
6	$-\frac{\pi}{2}$	2.7	0	$q_6 - \pi$
7	$-\frac{\pi}{2}$	2.7	3.6	$q_7 + \pi$

5.1.2 Sensors

There are a few options when choosing a suitable sensor to detect the position of the ball on the plate. A vast range of B&P applications makes use of vision sensors such as a camera which obtains the visual position of the ball using computer vision tools for filtering, processing, and detection of selected features [46]. This is a relatively straightforward solution that works in most cases, but its main disadvantage is the static position of the camera above the plate. This may not be suitable for this type of application because the robot can move freely not only in rotation but also in position which would require frequent calibration for any new change. A distortion of the lens and edge height differences of the plate in different inclinations can be also the source of errors for larger plate dimensions.

Another highly used approach is the touch panel on the plate - or in some cases the touch panel forming the plate itself [47]. A resistive touch screen for monitors is a great way to detect the position of the ball on the whole plate without any distortions based on plate inclination or its dimensions. The plate itself is a sensor and thus it travels with the movement of the robot and recalibration is not needed after changing the position of the plate in space. The only downside of a resistive touch panel is that the ball has to have a weight above the touch threshold of the panel. Balls with low mass thus cannot be used for this solution. A capacitive panel can of course be also used, but then the material of the ball is important instead of its mass.

A resistive touchscreen was chosen to serve as a sensor for this thesis as it is easy to use, implement and does not require additional static structures to hold other hardware. The plate used is a 322 x 247 x 2 mm glass plate with thin resistive touch foil and weighs approximately 420 g with plastic holders and flange mount. It is a classic analog 4-wire resistive touchscreen used as a spare part in monitors and displays (Fig 5.6).

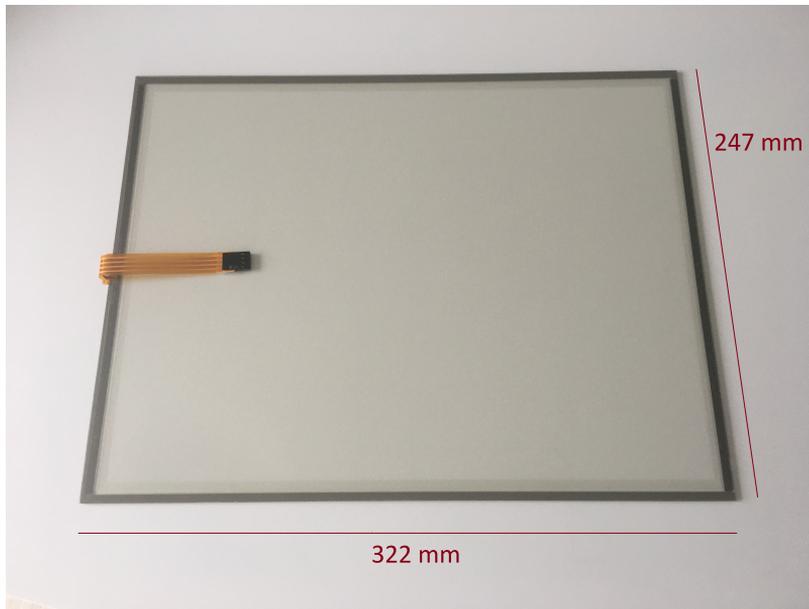


Fig. 5.6 Analog 4-wire resistive touchscreen

5.1.3 B&P Robotic System

The whole system is designed with the robot's 0.5 kg payload capacity in mind. The plate is therefore mounted directly on the flange of the robot to minimize the number of support materials and to keep the whole center of gravity as close to the flange as possible. The load diagram for this specific case can be seen in Fig. 5.7. The ball used is a steel bearing ball with a 25 mm diameter and weight of 64 g. It is obvious that the moving ball on the plate shifts the whole center of gravity based on its position, but this will be considered an external disturbance from the control system's perspective.

The configuration of the robot arm is also chosen carefully (Fig. 5.8) to relieve smaller joints (4-6) from larger forces.

The real setup of the B&P - robot system with plastic plate holders and testing fences with the ball placed in the middle is shown in Fig. 5.9.

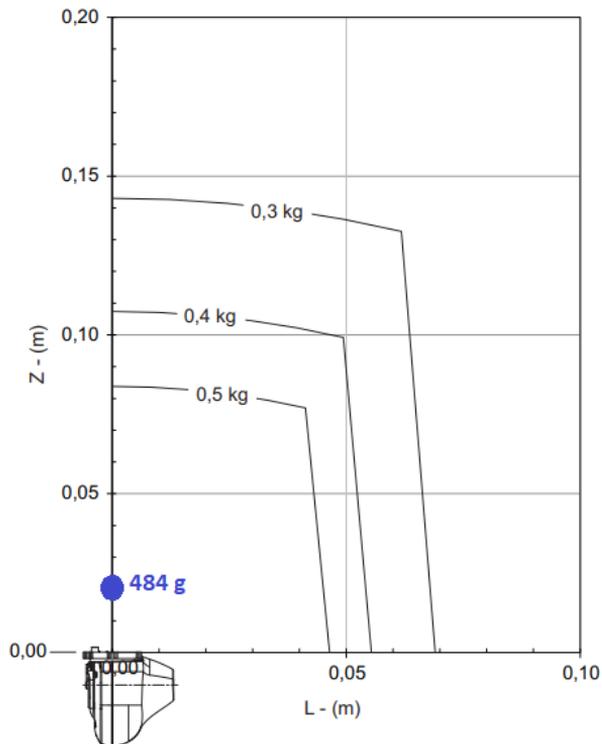


Fig. 5.7 Load diagram of the B&P setup for robot IRB 14000

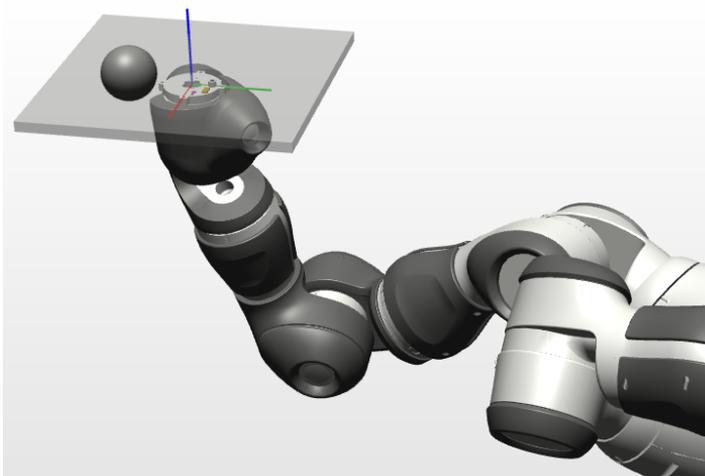


Fig. 5.8 Robot configuration in a default state

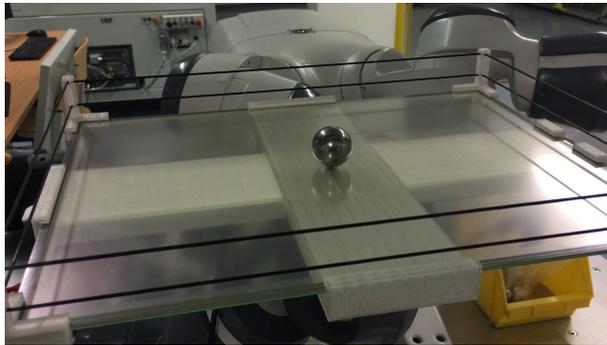


Fig. 5.9 B&P setup

The kinematic structure of the robot contains 7 actuators connected in series and although they do not contribute to the dynamics of needed rotation ($< 5^\circ$) equally, their overall contribution has to be taken into account. The transient function of the rotation of the robot's TCP (Tool Center Point) can be seen in Fig. 5.10 for a 10-degree step. A larger step change angle was chosen to better reflect the dynamics of the manipulator. The graph clearly shows 2nd-degree characteristics, but with relatively fast dynamics. The dynamics of the robot's movements are approximated with the 1st order transient function despite these measurements because it has much less impact on the whole system than its B&P part. This approximation thus trades the smaller complexity of the whole solution for the slight inaccuracy of the mathematical description of the system.

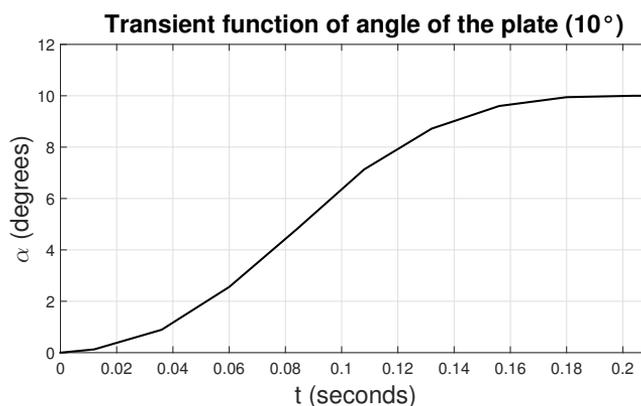


Fig. 5.10 Dynamics of the plate motion

5.2 Robot Guided Motion

A classic industrial robot is designed to fulfill standard point-to-point movements and they are carefully planned by a motion planner which also creates trajectories based on bezier curves. Almost all robotic applications take advantage of motion planning, but the motion planner should not be used to control robot movements based on fast-changing processes. In this case, the motion of the robot must be directly guided by an external sensor or device and the motion planner needs to be bypassed to keep the robot from moving to a previously planned point which is no longer valid because of the changed system outputs.

ABB robots can be equipped with the software option called EGM (Externally Guided Motion) which can be used for this purpose. It can communicate using an analog IO system or UDP protocol with data serialized using google protocol buffers (protobufs). EGM functionality has a latency of 4 – 20 *ms* and has 3 modes in which it can operate:

- Position streaming - used to stream robot position
- Path correction - used to correct planned trajectory
- Position guidance - used to directly guide the robot drive system

This thesis makes use of this EGM feature in position guidance mode using UDP protocol which is more complex to use and implement but offers better flexibility in terms of device choice that will run the designed controller.

5.3 Virtual Robot Scene

The initial design of the robot and B&P setup should be simulated, but standard simulation tools such as Matlab cannot be used to their full potential because the robot's internal structure and dynamics are not disclosed by the manufacturer.

A kinematic model might be enough for the initial simulation, but results would not fully reflect the real system and more importantly, the final program for the robot movements could not be transferred to the real counterpart. There are several options on the market, specifically for ABB robots. The most obvious candidate is the original software for robot programming and simulations from the manufacturer of the robot - RobotStudio [48][49]. Another option might be the widely used Process Simulate from Siemens [50] which can simulate robots of multiple brands and can be equipped with a robot core that can authentically simulate the robot in relation to the real system. RobotStudio was used to simulate the B&P model designed for this thesis Fig. 5.11.

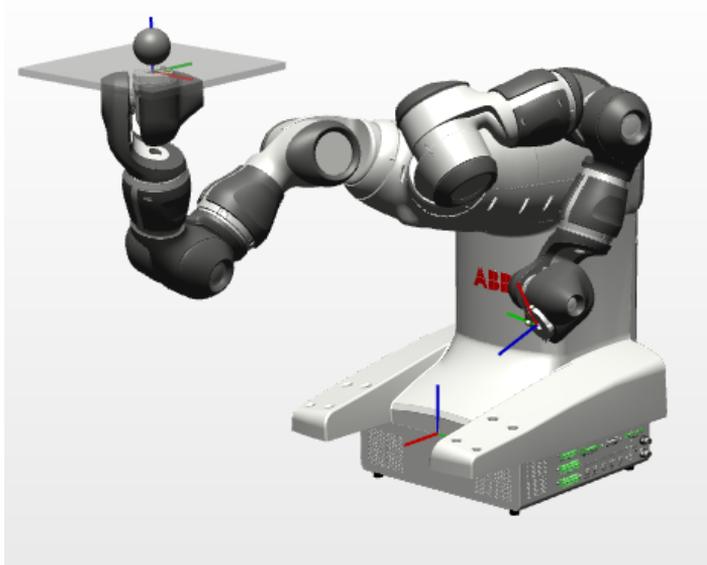


Fig. 5.11 RobotStudio setup of the B&P robotic system

It offers advanced tools for object positioning, measurements, and more importantly physics engine, thus achieving HIL (**H**ardware-**I**n-the-**L**oop) simulation standards. The physics engine used in RobotStudio is AGX Dynamics from Algorix and offers a multibody dynamics simulation library that can handle frictional contacts, which may be an interesting aspect for simulation purposes. Because of this engine, RobotStudio can set material properties for objects such as density, Young's modulus, Poisson's ratio, coefficient of restitution, and surface roughness (friction).

6 EXPERIMENTAL PART

The experimental part opens with the simulated system derivation, identification, and calculation of controller parameters. Simulation results are also presented along with a comparison of different controller design results in simulation. The real part of the experimental part consists of B&P system identification results and controller parameters calculation and it closes with results of control of the real system with different types of reference values to show the capabilities and limits of the designed controller.

6.1 Simulated System

The Ball & Plate simulated system in RobotStudio is used and evaluated in this chapter. Controller parameters are determined, implemented and different controller types are compared in the simulation environment. Different types of controllers are compared graphically and also the quality of control is calculated from the error and controller effort.

6.1.1 B&P Robotic System Identification

Dynamic parameters of the robot are not precisely known, thus identification of the system as a whole is the only option even in simulation (and can be called pseudo-identification). The whole robot is identified for the structure of the plant described in chapter 4.1.5 in equation (4.39). This equation approximates the whole motion structure by 1st order dynamic system which may seem an oversimplification to some extent, but as seen from the results this approximation is still valid for this case. The identification is based on a step response of the simulated system with the configuration of the manipulator shown in Fig. 5.11. Material parameters and characteristics of the plate and ball in the simulation were chosen to reflect real values as close as possible. The resulting step responses of the simulated system in one axis can be seen in Fig. 6.1.

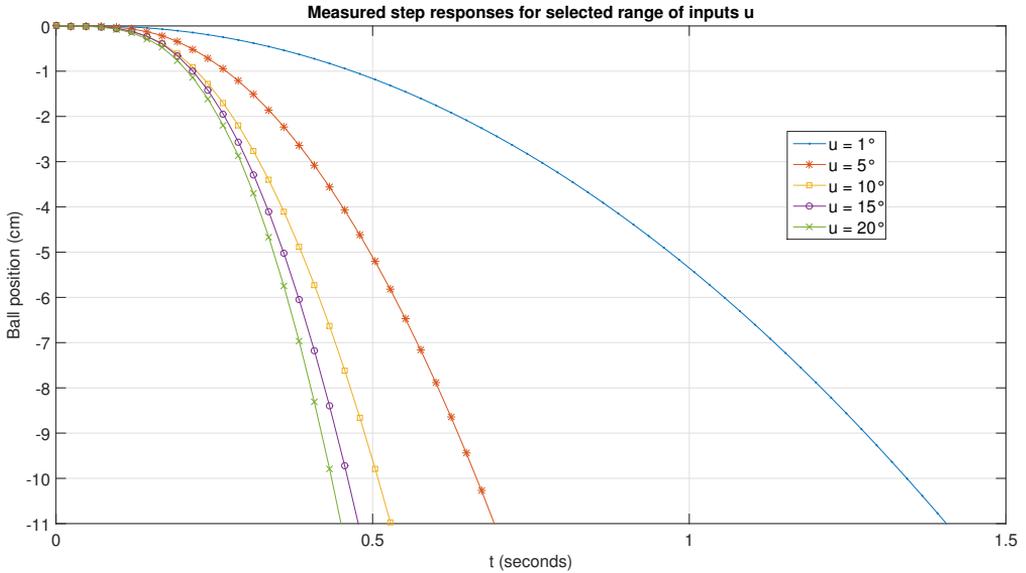


Fig. 6.1 Pseudo-identification of the system in simulation

These results were approximated by a least-square minimization method and Nelder-Mead simplex algorithm to find parameters of equation (4.39) according to these measurements. These parameters for 5-degree step change are shown in (6.1) and a plot of all parameters for step changes shown in Fig. 6.1 are shown in Fig. 6.2.

$$G(s) = \frac{K}{s^2(T_r s + 1)} = \frac{-0.1306}{s^2(0.1167s + 1)} \quad (6.1)$$

It can be seen the system significantly loses its linearity for plate angles greater than 15° which may be a result of the manipulator's large changes of multiple joints to achieve larger angles. This non-linearity can be however neglected because the plate will never move past a 10-degree inclination in the control cycle as it is unreasonably high for this specific use case.

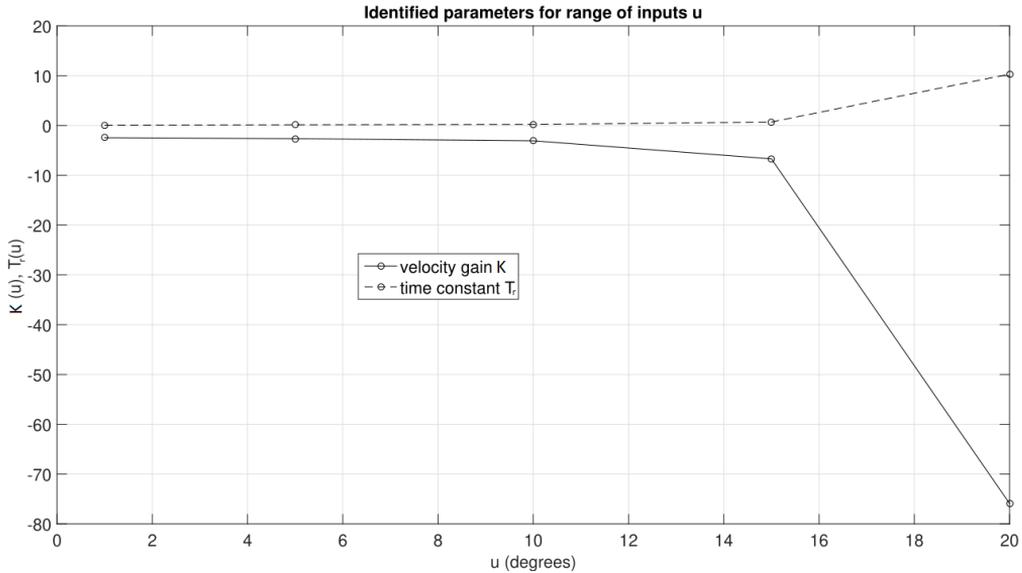


Fig. 6.2 Identified parameters for multiple step changes

6.1.2 Controller Parameters

A discrete version of equation (6.1) can be written based on the structure of equation (4.40) for a chosen time period of 0.05 s as shown in (6.2).

$$G(z^{-1}) = \frac{(-2.101z^{-1} - 7.577z^{-2} - 1.696z^{-3})10^{-5}}{1 - 2.652z^{-1} + 2.303z^{-2} - 0.6515z^{-3}} \quad (6.2)$$

This time period was chosen in accordance with the dynamics of the manipulator as seen in Fig. 5.10. It is expected that the angle of the plate will be bound by $\langle -2^\circ, 2^\circ \rangle$ range (higher angles have a small effect on the ball near the center of the plate and too large angles can introduce strong non-linearity to the ball movement such as jumping). Fig. 5.10 shows the manipulator can achieve these bounds within 0.05 s time. Lower time period values could introduce unwanted noise readings and as shown in Fig. 6.1 it is enough considering the dynamics of the ball even for larger angles. The 2 DoF polynomial LQ controller can be calculated as per chapters 4.2.1 and 4.2.2. Half of the characteristic polynomial D_{spf} is calculated by spectral factorization with penalization

constant $q_u = 10$ (6.3) from equation (4.58) for a more optimal control strategy and the second half D_{pp} (6.4) is selected by pole placement method with all three poles equal to $pp_{1,2,3} = 0.92$. The resulting characteristic polynomial $D(z^{-1}) = D_{spf}(z^{-1})D_{pp}(z^{-1})$ is shown in (6.5) and used for the calculation of controller parameters. These are shown in equations (6.6) and (6.7).

$$D_{spf}(z^{-1}) = 1 - 2.5408z^{-1} + 2.1251z^{-2} - 0.5826z^{-3} \quad (6.3)$$

$$D_{pp}(z^{-1}) = 1 - 2.7600z^{-1} + 2.5392z^{-2} - 0.7787z^{-3} \quad (6.4)$$

$$D(z^{-1}) = 1 - 5.3008z^{-1} + 11.677z^{-2} - 13.678z^{-3} + 8.9825z^{-4} - 3.1341z^{-5} + 0.4537z^{-6} \quad (6.5)$$

$$C_f(z^{-1}) = \frac{-0.007652}{1 - 1.6499z^{-1} + 0.6969z^{-2}} \quad (6.6)$$

$$C_b(z^{-1}) = \frac{-32.058 + 83.672z^{-1} - 71.644z^{-2} + 20.022z^{-3}}{1 - 1.6499z^{-1} + 0.6969z^{-2}} \quad (6.7)$$

Calculated parameters of the controller in matrix form are presented in equation (6.8) and can be directly used in equation (4.55).

$$x : \begin{bmatrix} r_0 \\ p1 \\ p2 \end{bmatrix} = \begin{bmatrix} -0.007652 \\ -1.6499 \\ 0.6969 \end{bmatrix}, \quad \begin{bmatrix} q0 \\ q1 \\ q2 \\ q3 \end{bmatrix} = \begin{bmatrix} -32.058 \\ 83.672 \\ -71.644 \\ 20.022 \end{bmatrix} \quad (6.8)$$

6.1.3 Results

Simulation measurements were done directly in RobotStudio which provides not only full virtualization of the robot but also simulates the ball position sensor and takes care of physics simulation. Ball positions were directly connected to the robot's control system which provides also tools for the implementation of the calculated controller. This system was able to successfully control the ball on the plate and handle any external disturbances. These disturbances were simulated by using RobotStudio's tool for object manipulation during simulation. Disturbances with random force and direction were introduced in the system, similar to pushing the ball in the real world. Results of this ball stabilization control are presented in Fig. 6.3 for x coordinate of the ball and in Fig. 6.4 for y coordinate.

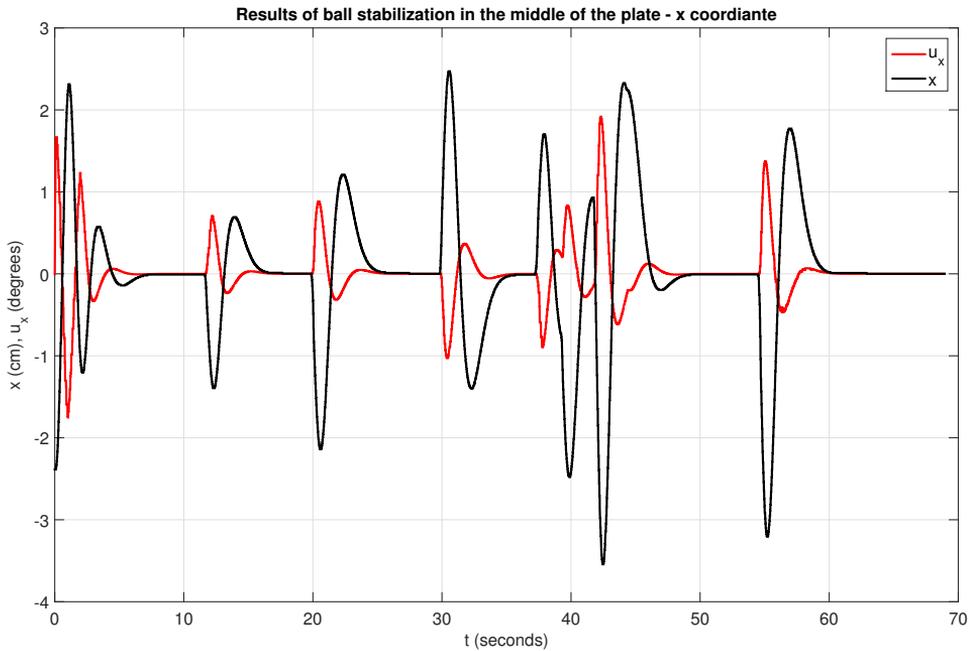


Fig. 6.3 Simulation results for x coordinate

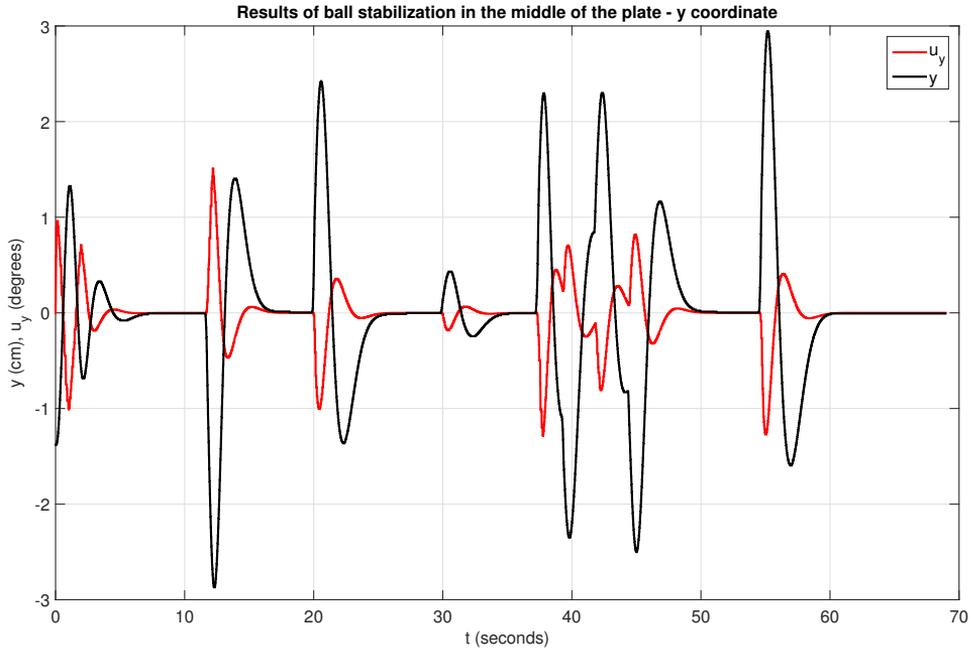


Fig. 6.4 Simulation results for y coordinate

6.1.4 Comparison

So far only one specific method was used to design the controller with a smaller impact on the mechanics of the robot in mind, but this reasoning should be supported by relevant data. This section compares different methods of controller design and their usability for the purpose of this thesis. Compared are optimal LQ polynomial design for 2 DoF controller structure (described in this thesis and labeled as LQ opt in figures), standard discrete PD controller (designed using Naslin's method [51] and labeled PD in figures) and state-space LQR controller (labeled SS LQR in figures). These types of controllers are commonly used for this type of problem and were simulated in Matlab/Simulink disregarding their continuous or discrete character. Controller effort is also shown in the results as it is an important criterion for showing the advantages and disadvantages of each method. The simulation was conducted for reference value changing as step (Fig. 6.5 and Fig. 6.6), sequence (Fig. 6.7 and Fig. 6.8), ramp (Fig. 6.9 and Fig. 6.10), and harmonic (Fig. 6.11 and Fig. 6.12) processes and only in one

positional coordinate. All controllers were designed for a step change to better compare their results.

Two quality criteria are presented for each control result - a sum of squared errors (6.9) to compare controllers' performance and a sum of squared outputs (6.10) to compare controllers' effort. Results of these comparisons are presented in tables 6.1-6.4 and the best results are shown in bold.

$$S_e = \frac{1}{N} \sum_{k=1}^N e^2(k) \quad [cm^2] \quad (6.9)$$

$$S_u = \frac{1}{N} \sum_{k=1}^N u^2(k) \quad [deg^2] \quad (6.10)$$

Results show that unfiltered reference values in LQR and PD control may cause a large controller effort which is not appropriate nor achievable in this specific robotic setup. The advantage of the 2 DoF structure of the controller is quite obvious, although it reduces the quality of the control slightly. An interesting fact in the step (or sequence) change is that PD and LQR controllers are much faster in reaching the reference value at the beginning of the control action, but the settling time of all three controllers is basically the same.

These results show the 2 DoF controller (or basically the one with filtered reference value) is a solid choice for processes requiring lower overall controller effort. They also show the polynomial input-output control can compete to a certain extent with other more broadly used methods, although errors in the system are not simulated in this case.

PD controller is quite efficient for ramp and harmonic changes of reference value because it does not contain any feed-forward filtering. LQ controller is specifically designed for a step change in these comparisons so linear or harmonic change causes a phase shift of the controller value compared to the desired reference.

Tab. 6.1 Quality of control for a step change

	LQR	LQ opt	PD
S_e	0.0080	0.0229	0.0081
S_u	2.4203	0.2963	5.4281

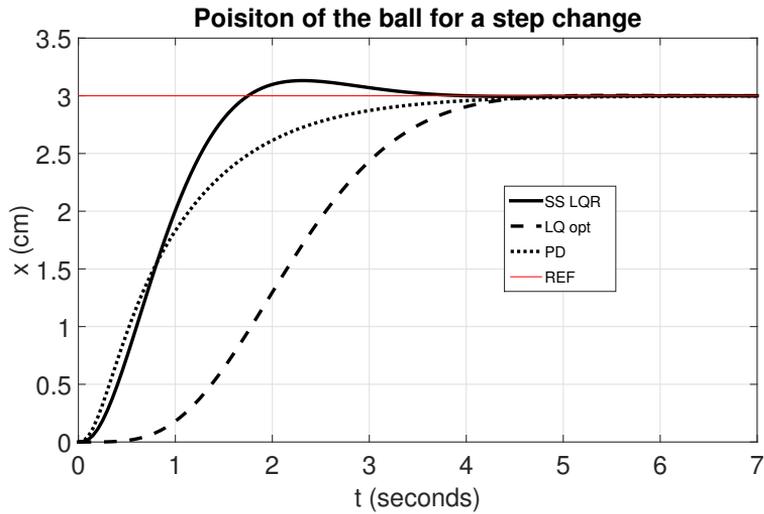


Fig. 6.5 Position of the ball for a step change

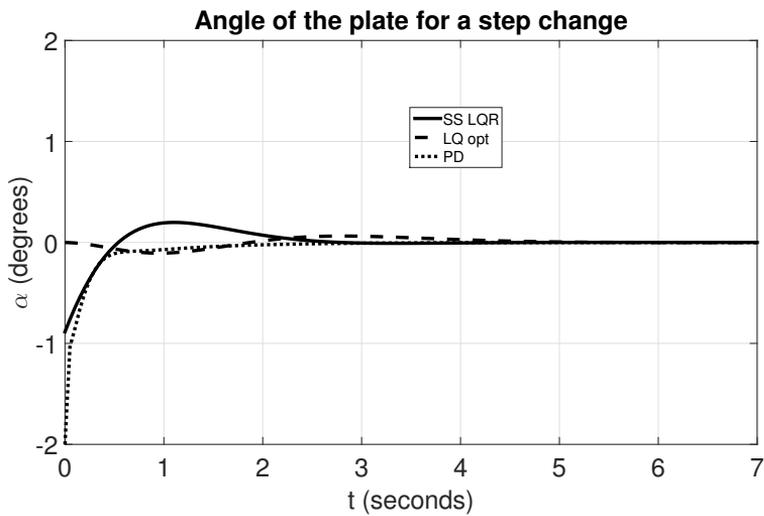


Fig. 6.6 Angle of the plate for a step change

Tab. 6.2 Quality of control for a sequence change

	LQR	LQ opt	PD
S_e	0.0475	0.1348	0.0548
S_u	13.9880	1.4078	16.9862

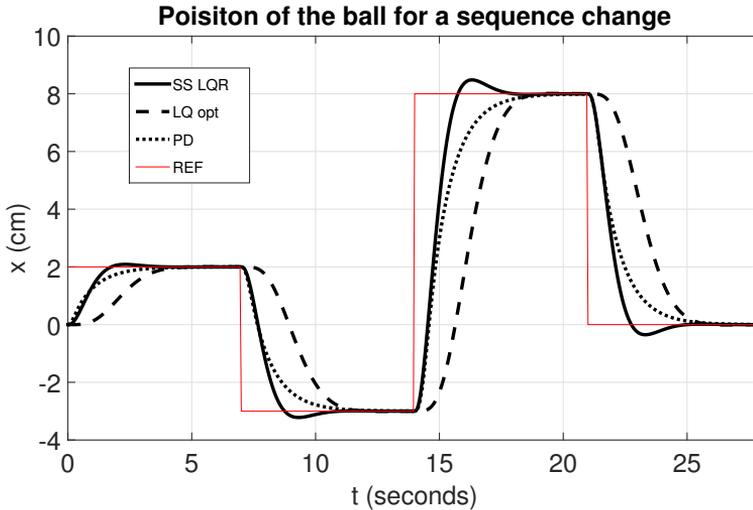


Fig. 6.7 Position of the ball for a sequence change

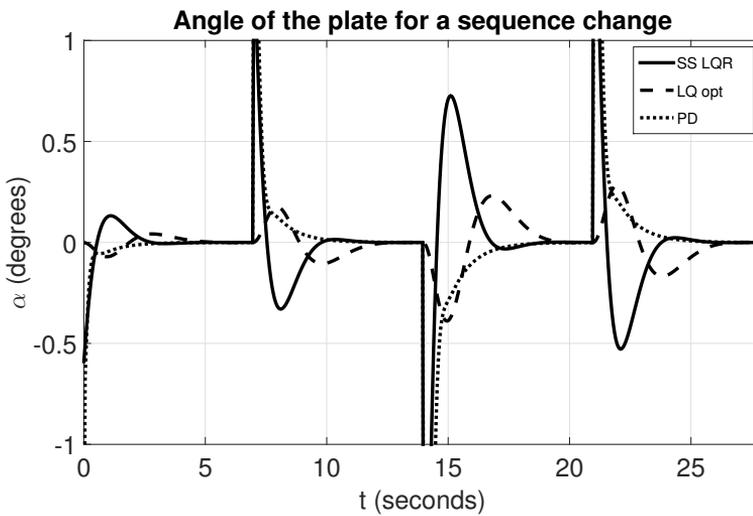


Fig. 6.8 Angle of the plate for a sequence change

Tab. 6.3 Quality of control for a linear change

	LQR	LQ opt	PD
S_e	0.0010	0.0076	$\sim \mathbf{0}$
S_u	0.1292	0.1248	0.1637

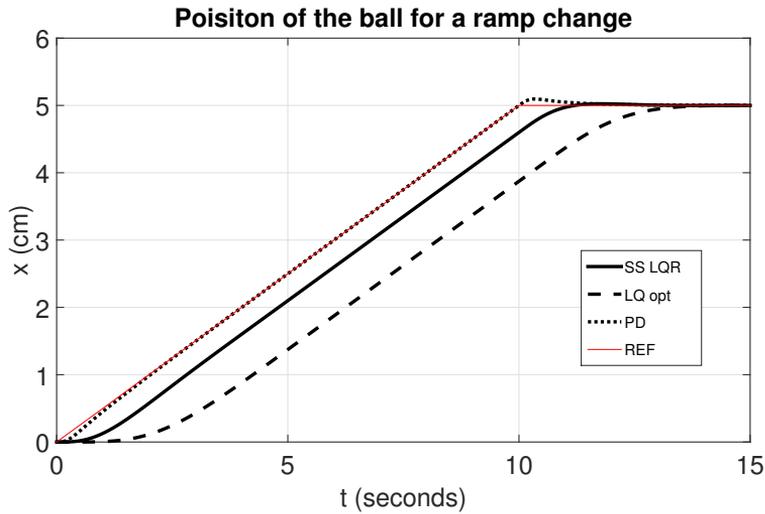


Fig. 6.9 Position of the ball for a linear change

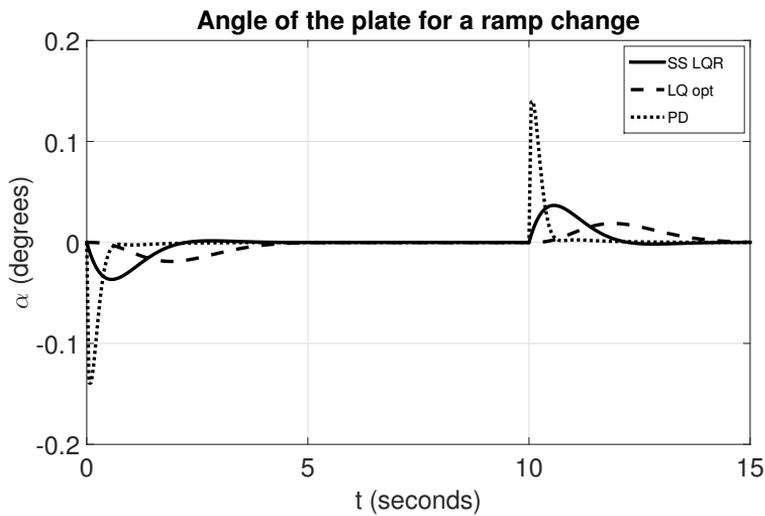


Fig. 6.10 Angle of the plate for a linear change

Tab. 6.4 Quality of control for a harmonic change

	LQR	LQ opt	PD
S_e	0.0022	0.0156	$\sim \mathbf{0}$
S_u	0.0450	0.0490	0.0625

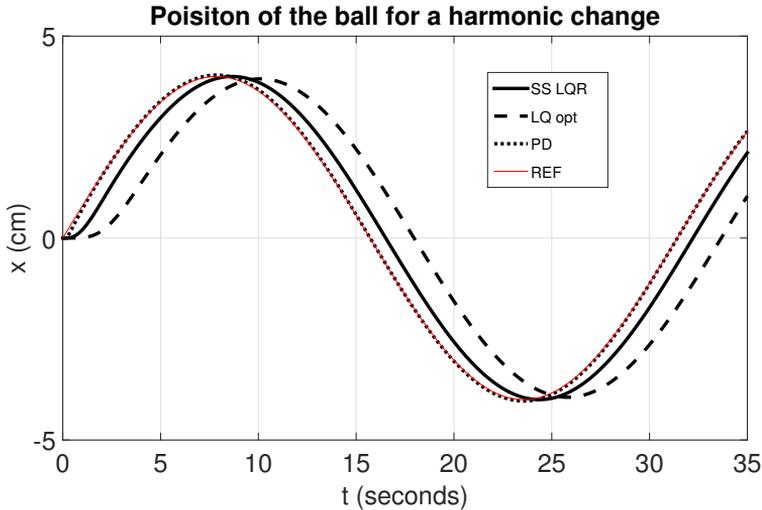


Fig. 6.11 Position of the ball for a harmonic change

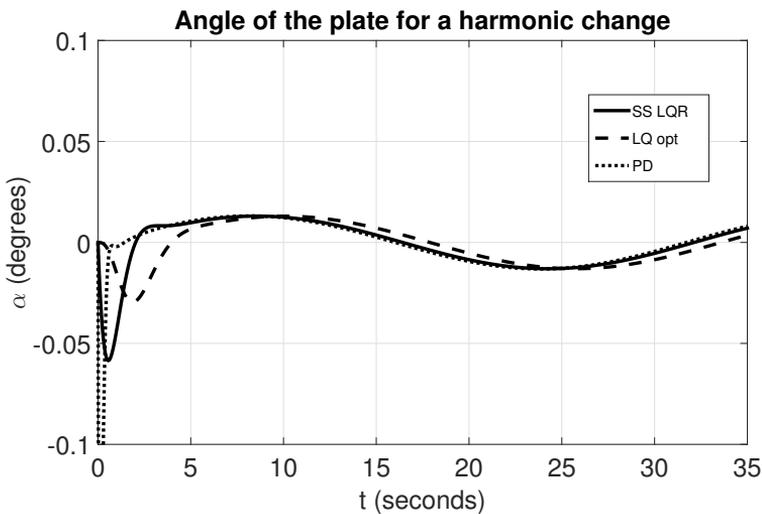


Fig. 6.12 Angle of the plate for a harmonic change

6.1.5 Analysis of Sensitivity to Model Errors

Uncertainties in the model caused by linearization, measurement errors, and process setup need to be considered and the robustness of the designed control strategy should be thus analyzed. Four possible errors of the model in equation (6.1) were taken into account for a step change - error of the gain K by $\pm 30\%$ (Fig. 6.13), error of time constant T_r by $\pm 90\%$ (Fig. 6.14), error of both of these parameters (linearly at the same time) shows Fig. 6.15 and addition of the transport delay of the controller output (caused by random delays in the robot system) from 10% to 300% of the sampling time for which the controller was designed (0.05 s) is shown in Fig. 6.16. The biggest impact has the gain K which is bound to $\pm 30\%$ and bigger changes would cause instability of the system eventually. Very interesting is the analysis of the controller output transport delay. It shows the designed controller is relatively robust against changes in the reaction of the controller system.

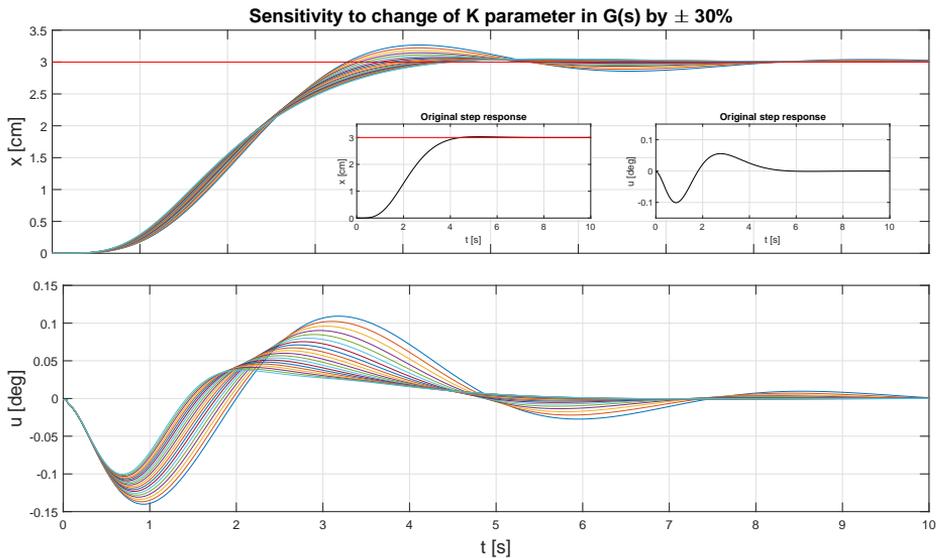


Fig. 6.13 Sensitivity to change of K parameter

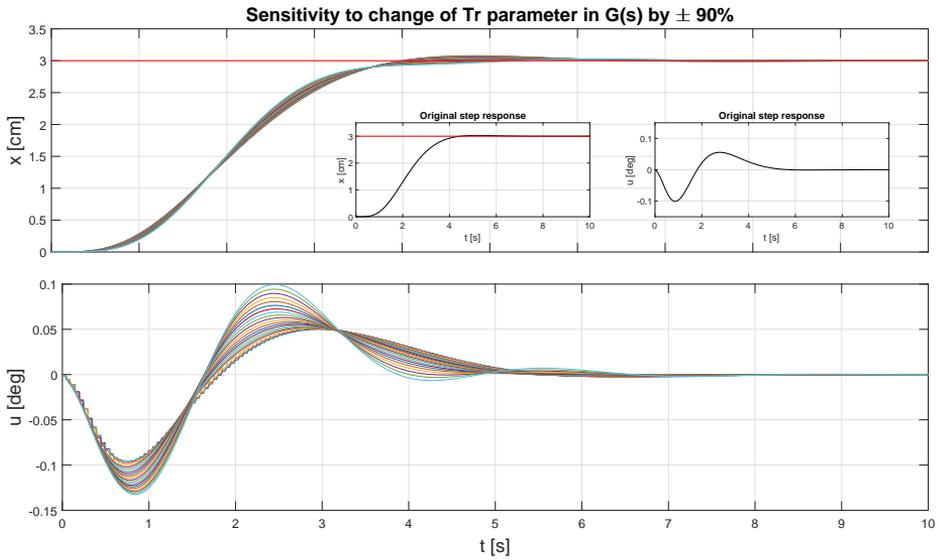


Fig. 6.14 Sensitivity to change of Tr parameter

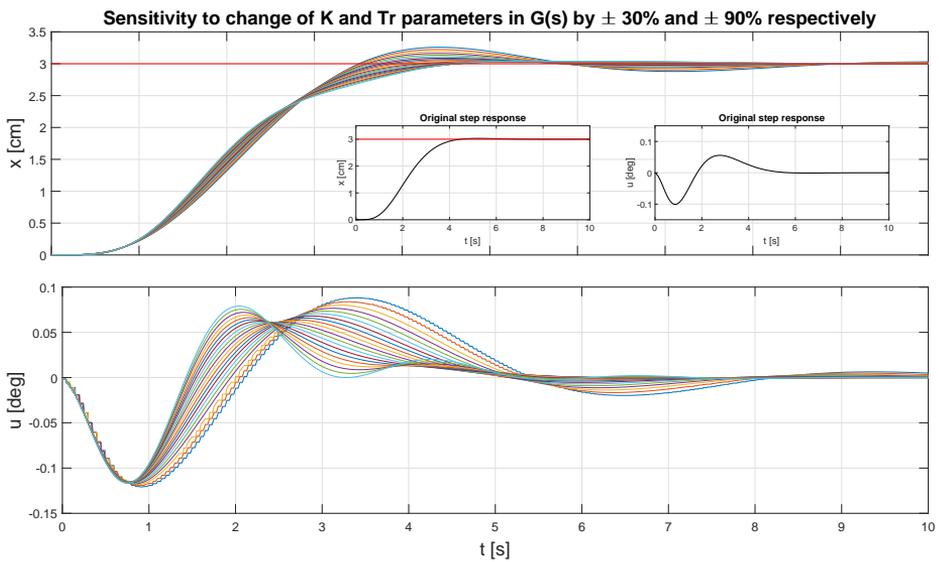


Fig. 6.15 Sensitivity to change of both K and Tr parameters

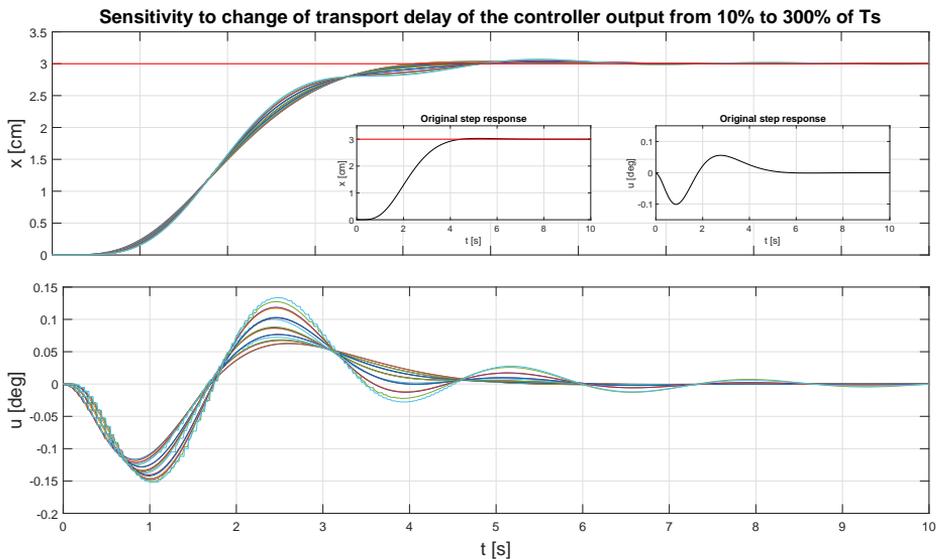


Fig. 6.16 Sensitivity to change of transport delay of the controller output

The parameters of the robotic manipulator can change slightly in time due to increased friction in gearboxes, but they can also change with different loads affecting the manipulator. The stability of the controller can not be compromised, although the quality of control will be obviously worse than in the original design. The robot itself is also not bound to execute desired controller effort in a precise time interval which may cause problems because the controller is designed for a specific time sampling period. It is shown that approach in this thesis is also robust against the increased transport delay of the motion system.

This analysis provides a good background for the deployment into a real system and shows that many uncertainties of the system and approximations of the model should not cause instability of the overall system and that the robustness of this control strategy is within a safe level for such a fast and unstable system. A thorough and deep analysis of the robustness was not conducted and only the sensitivity of the model to various errors is presented, which is enough to prove the claims of stability and robustness of the controller.

6.2 Real System

The ball & Plate system mounted on the robotic manipulator is presented in this chapter, starting with its identification and controller design. Various reference value signals are tested and their results are shown for both axes and also in the x-y plane. The real system is not symmetric for x and y coordinates and thus requires separate calculations for both coordinates, although the differences are not large and could be easily simplified to errors in the model and thus disturbances in the system.

6.2.1 B&P Robotic System Identification

Identification of the real system was conducted in a similar manner as for the virtual one, but multiple measurements were taken to decrease the error (Fig. 6.17 showing measurements for 2-degree step change). Each measurement was identified and the resulting coefficients of equation (4.39) were averaged to obtain a single transfer function of the system shown in equation (6.11) for the x coordinate and equation (6.12) for the y coordinate (they are not exactly symmetric in the real system). Response of this function is directly plotted over measurements for 2-degree step change in Fig. 6.18, but was obtained from averaged coefficients of measurements for different step changes also. The correlation between these individually measured coefficients is shown in Fig. 6.19 for the x coordinate and in Fig. 6.20 for y coordinate. This also shows how closely related are coefficients of unstable aperiodic systems with different combinations of values providing similar responses.

$$G_x(s) = \frac{K}{s^2(T_r s + 1)} = \frac{0.8306}{s^2(0.4687s + 1)} \quad (6.11)$$

$$G_y(s) = \frac{K}{s^2(T_r s + 1)} = \frac{0.9168}{s^2(0.4108s + 1)} \quad (6.12)$$

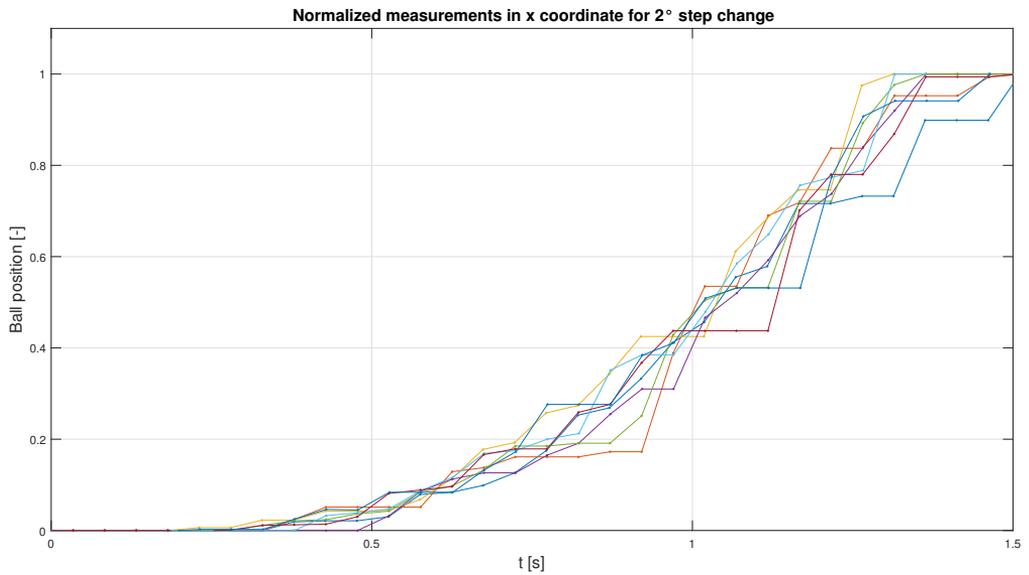


Fig. 6.17 Measurements for identification of the system

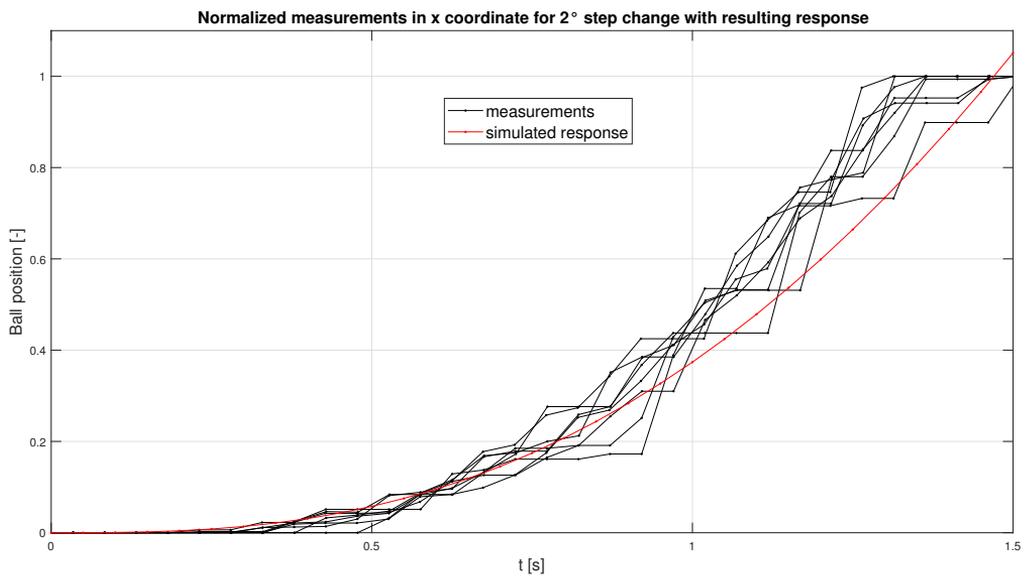
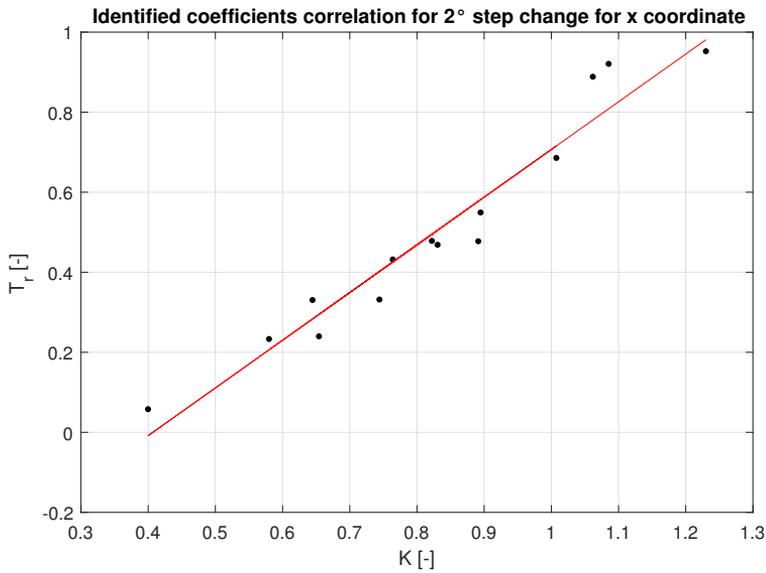
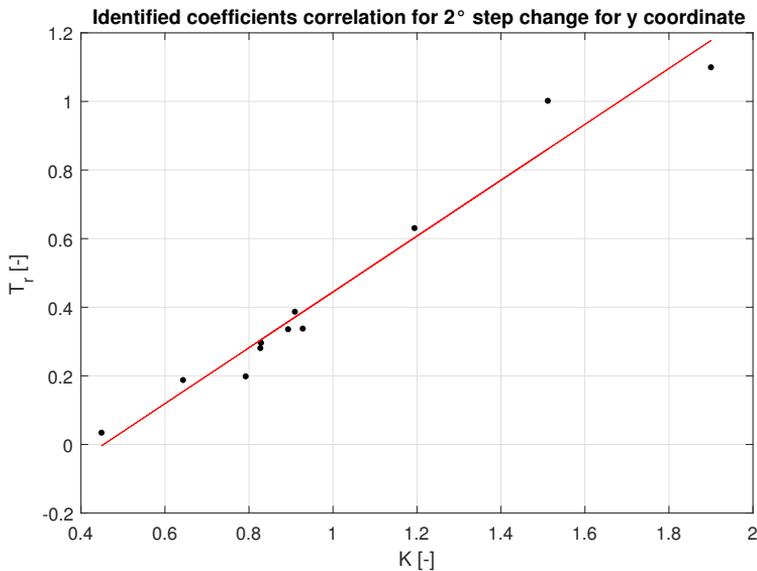


Fig. 6.18 Measurements with the response of averaged coefficients

Fig. 6.19 Correlation of identified coefficients for x coordinateFig. 6.20 Correlation of identified coefficients for y coordinate

6.2.2 Controller Parameters

Similarly to simulation chapter 6.1.2, equations (6.11), (6.12) are discretized based on the (4.40) for a time period of 0.05 s and shown in (6.13) and (6.14).

$$G_x(z^{-1}) = \frac{(3.596z^{-1} + 14.01z^{-2} + 3.409z^{-3})10^{-5}}{1 - 2.899z^{-1} + 2.798z^{-2} - 0.8988z^{-3}} \quad (6.13)$$

$$G_y(z^{-1}) = \frac{(4.512z^{-1} + 17.51z^{-2} + 4.245z^{-3})10^{-5}}{1 - 2.885z^{-1} + 2.771z^{-2} - 0.8854z^{-3}} \quad (6.14)$$

The rationale behind the choice of time period is described in chapter 6.1.2, but to sum up, 0.05 s was chosen based on the combination of dynamics of the manipulator and the B&P system (Fig. 5.6 and Fig. 6.1 respectively).

Chapters 4.2.1 and 4.2.2 describe controller design where the parameters of controllers are chosen the same as in chapter 6.1.2 - penalization constant of spectral factorization $q_u = 10$ and three placed poles $pp_{1,2,3} = 0.92$. Results are shown for both coordinates - polynomial from spectral factorization in (6.15) and (6.16), pole-placed polynomial in (6.17), final characteristic polynomial $D(z^{-1}) = D_{spf}(z^{-1})D_{pp}(z^{-1})$ in (6.18) and (6.19) and resulting controllers in (6.20)-(6.23).

$$D_{spf_x}(z^{-1}) = 1 - 2.7924z^{-1} + 2.6004z^{-2} - 0.8074z^{-3} \quad (6.15)$$

$$D_{spf_y}(z^{-1}) = 1 - 2.7846z^{-1} + 2.5849z^{-2} - 0.7998z^{-3} \quad (6.16)$$

$$D_{pp}(z^{-1}) = 1 - 2.7600z^{-1} + 2.5392z^{-2} - 0.7787z^{-3} \quad (6.17)$$

$$D_x(z^{-1}) = 1 - 5.5524z^{-1} + 12.847z^{-2} - 15.854z^{-3} + 11.006z^{-4} - 4.0751z^{-5} + 0.6287z^{-6} \quad (6.18)$$

$$D_y(z^{-1}) = 1 - 5.5446z^{-1} + 12.810z^{-2} - 15.784z^{-3} + 10.939z^{-4} - 4.0437z^{-5} + 0.6228z^{-6} \quad (6.19)$$

$$C_{f_x}(z^{-1}) = \frac{0.001462}{1 - 1.6542z^{-1} + 0.7001z^{-2}} \quad (6.20)$$

$$C_{b_x}(z^{-1}) = \frac{17.438 - 49.789z^{-1} + 47.356z^{-2} - 15.004z^{-3}}{1 - 1.6542z^{-1} + 0.7001z^{-2}} \quad (6.21)$$

$$C_{f_y}(z^{-1}) = \frac{0.000975}{1 - 1.6598z^{-1} + 0.7039z^{-2}} \quad (6.22)$$

$$C_{b_y}(z^{-1}) = \frac{12.691 - 36.115z^{-1} + 34.224z^{-2} - 10.799z^{-3}}{1 - 1.6598z^{-1} + 0.7039z^{-2}} \quad (6.23)$$

Controller parameters in matrix form are presented in (6.24) and (6.25).

$$x : \begin{bmatrix} r_0 \\ p1 \\ p2 \end{bmatrix} = \begin{bmatrix} 0.001462 \\ -1.6542 \\ 0.7001 \end{bmatrix}, \quad \begin{bmatrix} q0 \\ q1 \\ q2 \\ q3 \end{bmatrix} = \begin{bmatrix} 17.438 \\ -49.789 \\ 47.356 \\ -15.004 \end{bmatrix} \quad (6.24)$$

$$y : \begin{bmatrix} r_0 \\ p1 \\ p2 \end{bmatrix} = \begin{bmatrix} 0.000975 \\ -1.6598 \\ 0.7039 \end{bmatrix}, \quad \begin{bmatrix} q0 \\ q1 \\ q2 \\ q3 \end{bmatrix} = \begin{bmatrix} 12.691 \\ -36.115 \\ 34.224 \\ -10.799 \end{bmatrix} \quad (6.25)$$

6.2.3 Results

Two approaches were implemented for controlling the manipulator's movement - sending angles to the robot's motion planner and bypassing the motion planner by sending desired angles directly to the motion system. All results for ball positions are normalized to compensate for unequal dimensions of the plate (see Fig. 5.6) during comparisons. Both approaches are presented below for the stabilization process.

Direct Method (using the motion planner)

The first option is easier to implement and is a standard programming method for robotic manipulators. The robot has prepared routines for communication with its motion planner in the form of standard linear or joint motion commands. The motion planner reads these commands, interpolates the path, and plans the movement accordingly. This has a clear setback in added computation overhead and responsiveness because once the motion is planned it has to be executed which goes directly against the idea of rapidly changing values from the controller. Results of ball stabilization in the center after an initial random disturbance are shown in Fig. 6.21 and Fig. 6.22. These results show pretty poor stabilization because the motion planner is not able to keep up with the controller and introduces unexpected (and random) time delay into the system in tenths of a second. Also, controllers for these measurements were calculated with placed poles closer to 1 ($pp_{1,2,3} = 0.97$) to make the controller effort smaller which kept the control process at least stable.

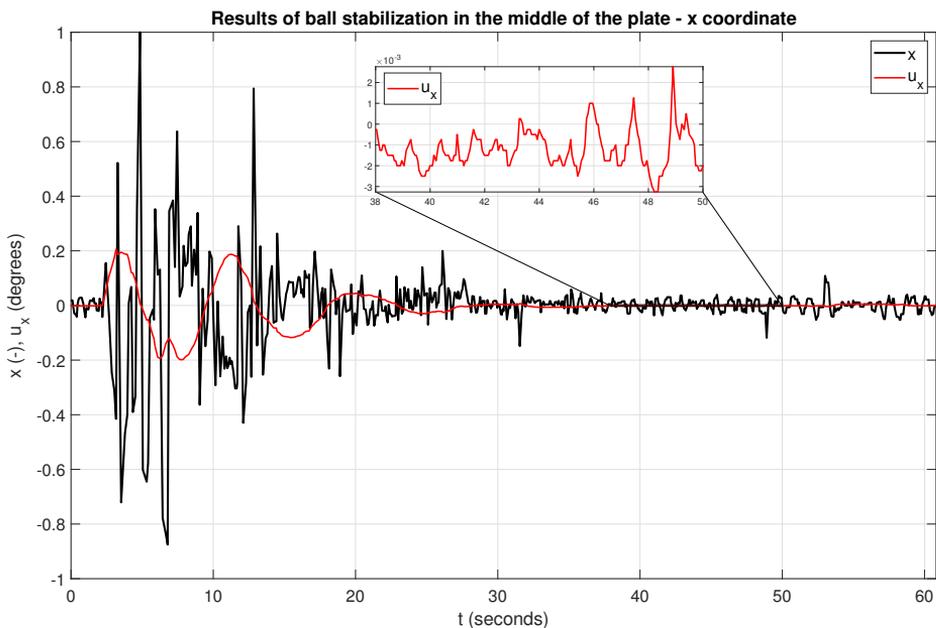


Fig. 6.21 Control results for x coordinate with motion planner

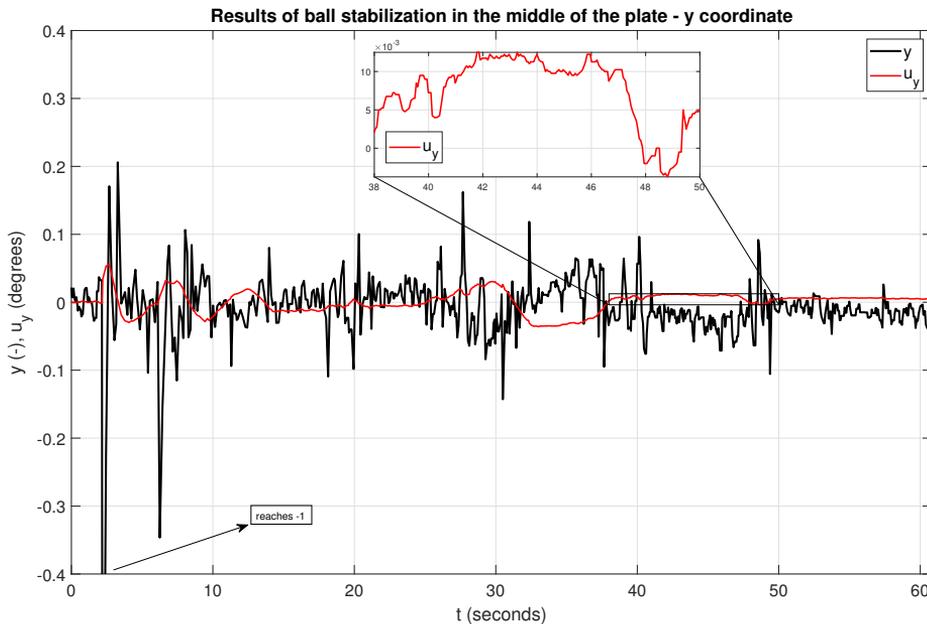


Fig. 6.22 Control results for y coordinate with motion planner

Robot Guided Method (bypassing the motion planner)

The second option, described in chapter 5.2, is faster as it bypasses the motion planner and is also more responsive to sudden changes because it does not follow a point-to-point strategy. Results for ball stabilization can be seen in Fig. 6.23 and Fig. 6.24 for x and y coordinates respectively and in Fig. 6.25 which shows the position of the ball on the plate in both coordinates. Multiple disturbances were introduced in the form of random impulse force applied externally to the ball and graphs clearly show when in time was the force applied. Desired reference value was 0° (in the center of the plate), so the stabilization and disturbance rejection can be clearly shown.

Graphs show the controller is able to respond to disturbances and stabilize the ball in the center in 3-5 seconds depending on the magnitude of the external force applied. Magnitudes of forces (or rather the deflection of the position of the ball) can be seen in the x-y plot (Fig. 6.25) as diagonal peaks of the ball's position.

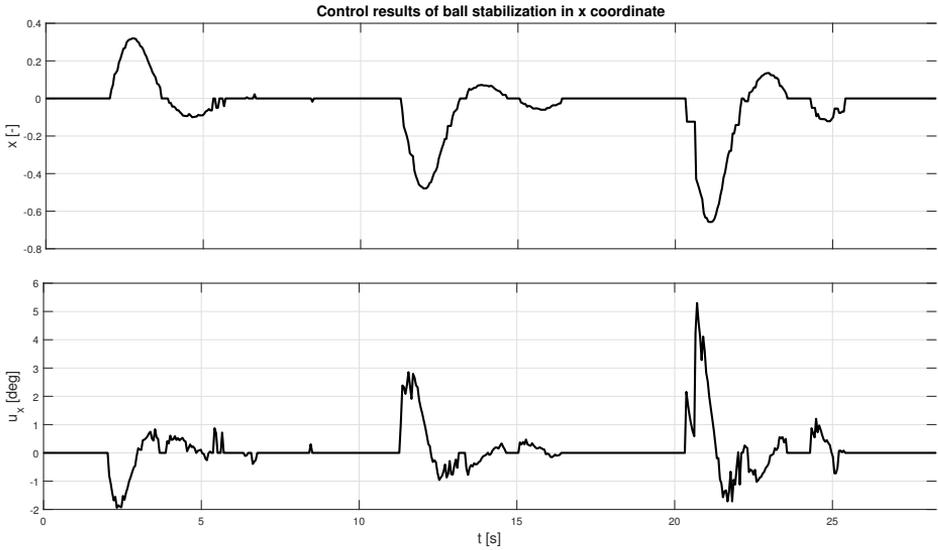


Fig. 6.23 Control results for stabilization in x coordinate

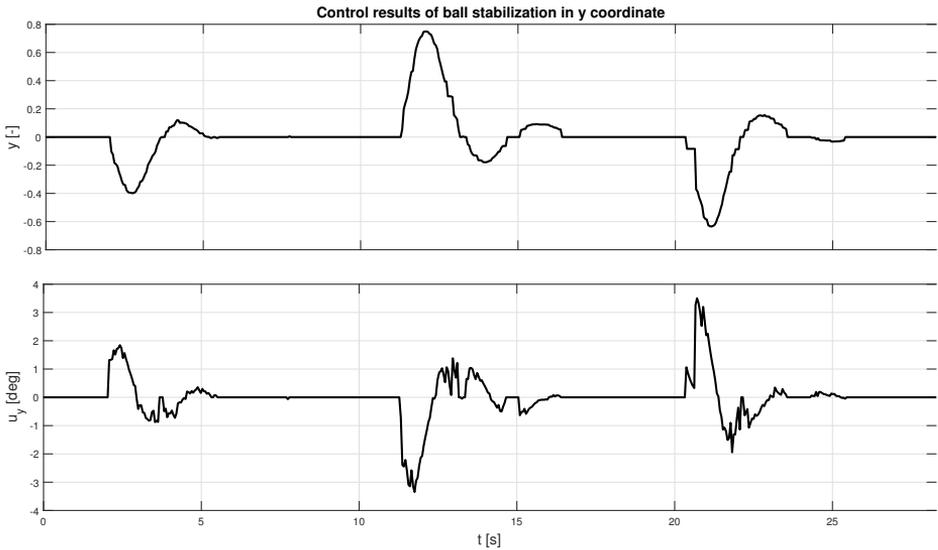


Fig. 6.24 Control results for stabilization in y coordinate

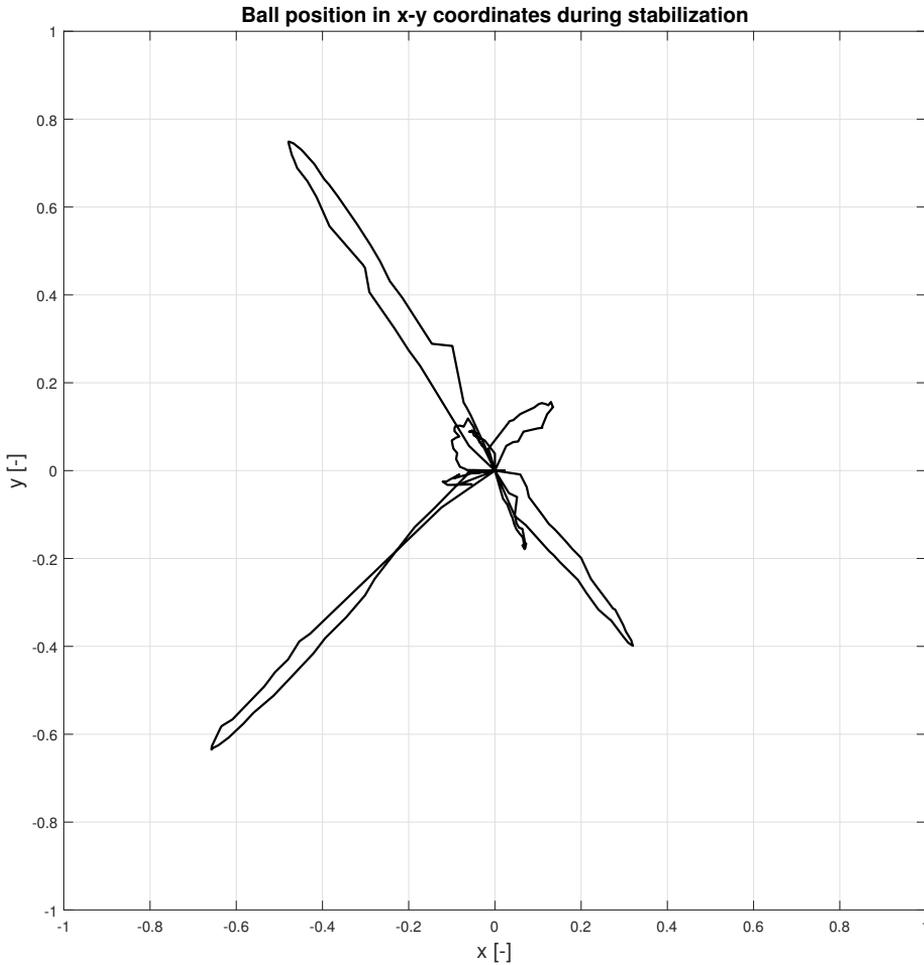


Fig. 6.25 Control results for stabilization in x-y plane

Another set of tests was a harmonic change of the reference value. The controller was not designed for harmonic change and its feed-forward part (responsible for reference tracking) is not able to track the desired input. This causes a shift in phase and a significant reduction of amplitude. This reduction is increasing with the rising frequency of the reference harmonic signal. Fig. 6.26-6.31 show tracking of harmonic reference value for frequencies 0.15 Hz and 0.25 Hz where the reference value needs to have amplitude out of bounds of the plate for the ball following a reasonable path.

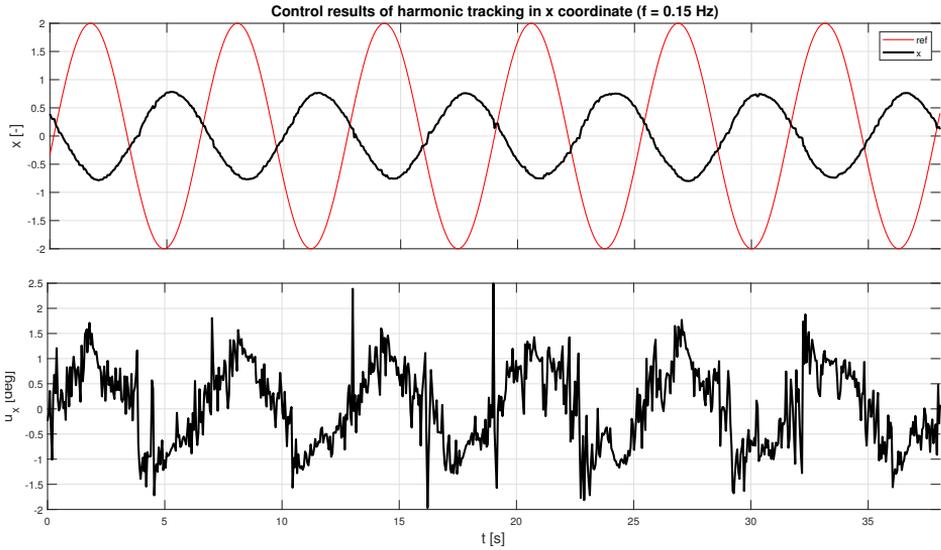


Fig. 6.26 Control results for harmonic tracking in x coordinate

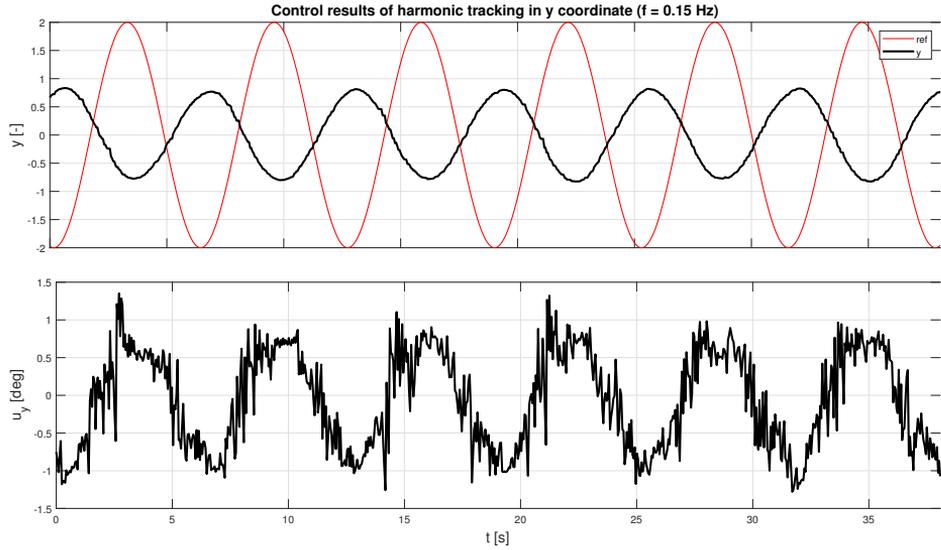


Fig. 6.27 Control results for harmonic tracking in y coordinate



Fig. 6.28 Control results for harmonic tracking in the x-y plane

Results for a frequency of 0.15 Hz show quite a large phase shift and amplitude reduction compared to the desired reference value. This effect is getting worse for higher frequencies and the increase by only 0.1 Hz (+67 %) has a very high impact on the resulting amplitude of the ball which is shown in the following figures. The reference value has to be even set out of the bounds of the plate to move the ball farther from the center and a great difference can be especially seen in Fig. 6.31.

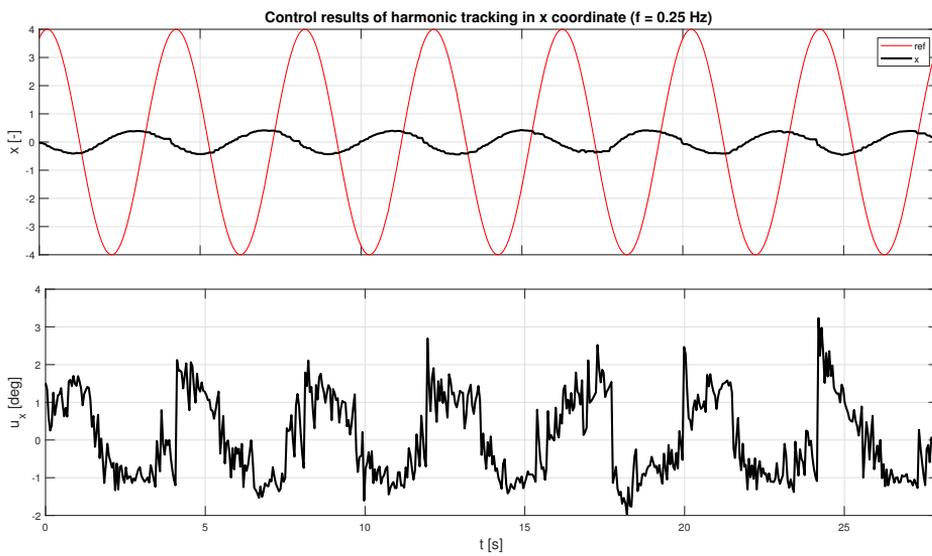


Fig. 6.29 Control results for harmonic tracking in x coordinate

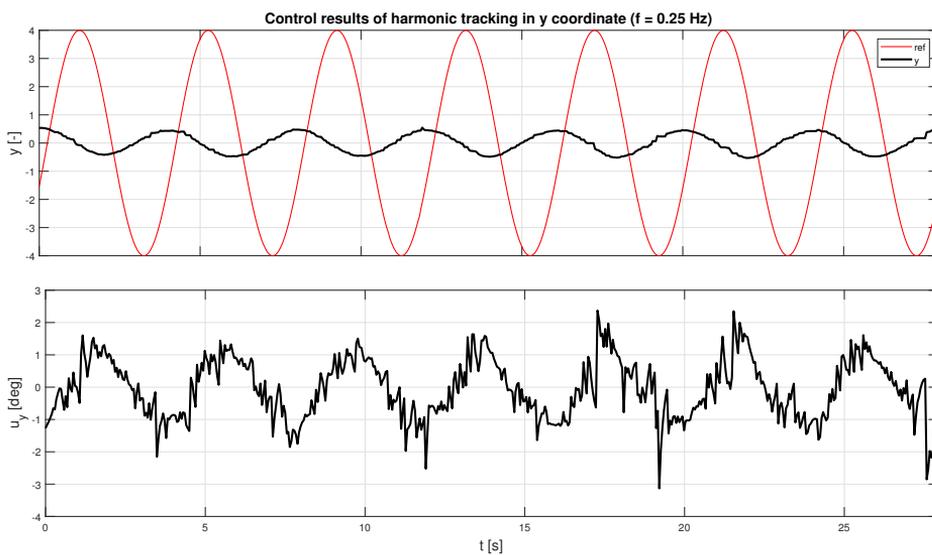


Fig. 6.30 Control results for harmonic tracking in y coordinate

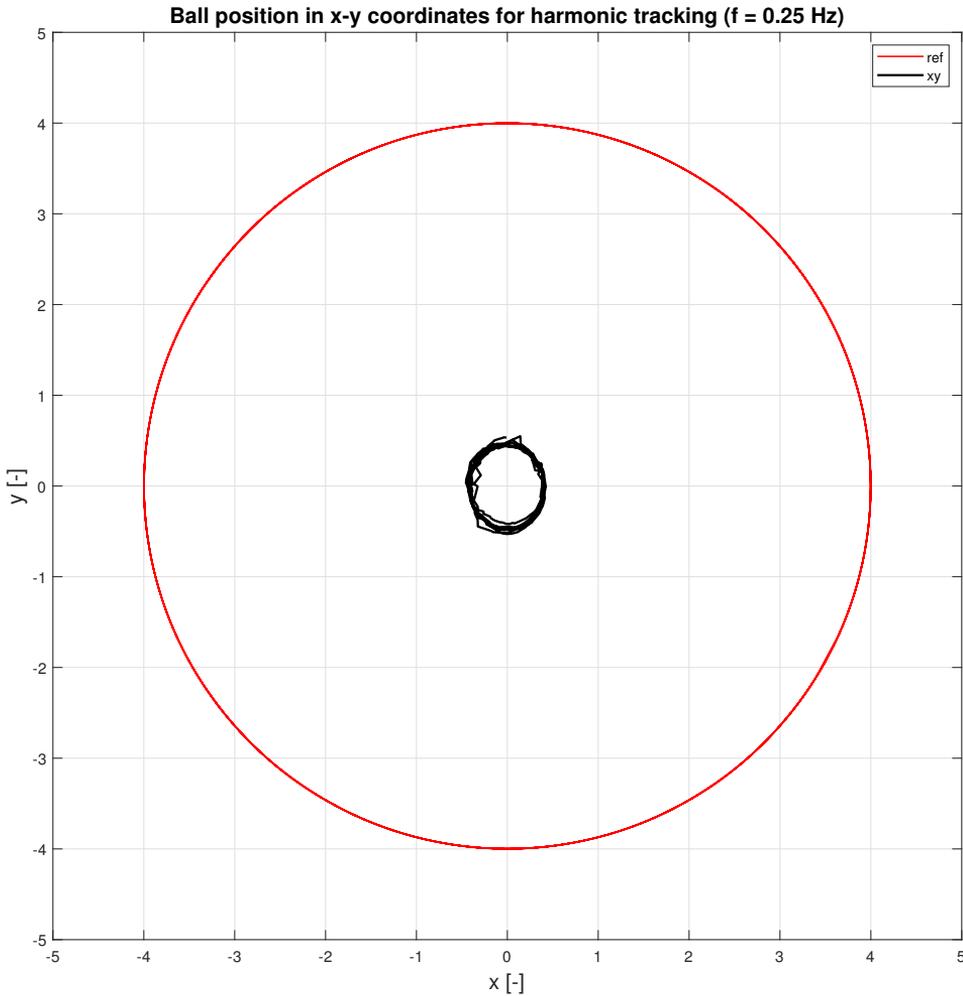


Fig. 6.31 Control results for harmonic tracking in the x-y plane

The controller was thus designed for harmonic change reference value, which needs an additional calculation of the nominator of the feed-forward part of the controller ($R(z^{-1})$), as described in equation (4.52) and more closely in [36]. Parameters of $R(z^{-1})$ are calculated based on equation (6.26), where $D_w(z^{-1})$ is the denominator of the reference value expressed in the polynomial form ($D_w = (1 - 2z^{-1}\cos\omega_0 + z^{-2})$) for harmonic change with angular frequency ω_0) and $S(z^{-1})$ is the remainder not used in any further calculations (more in [36]). This leads to a similar system of linear equations as in equation (4.59) and

can be solved for unknown $R(z^{-1})$ and $S(z^{-1})$ (although $S(z^{-1})$ is not used any further). The controller designed for a targeted frequency of 0.25 Hz was thus calculated and implemented as seen in equations (6.27)-(6.30).

$$D(z^{-1}) = D_w(z^{-1})S(z^{-1}) + B(z^{-1})R(z^{-1}) \quad (6.26)$$

$$C_{f_x}(z^{-1}) = \frac{-0.07329 + 0.07198z^{-1}}{1 - 1.6542z^{-1} + 0.7001z^{-2}} \quad (6.27)$$

$$C_{f_y}(z^{-1}) = \frac{-0.06677 + 0.06608z^{-1}}{1 - 1.6598z^{-1} + 0.7039z^{-2}} \quad (6.28)$$

$$x : \begin{bmatrix} r_0 \\ r_1 \\ p_1 \\ p_2 \end{bmatrix} = \begin{bmatrix} -0.07329 \\ 0.07198 \\ -1.6542 \\ 0.7001 \end{bmatrix}, \quad \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = \begin{bmatrix} 17.438 \\ -49.789 \\ 47.356 \\ -15.004 \end{bmatrix} \quad (6.29)$$

$$y : \begin{bmatrix} r_0 \\ r_1 \\ p_1 \\ p_2 \end{bmatrix} = \begin{bmatrix} -0.06677 \\ 0.06608 \\ -1.6598 \\ 0.7039 \end{bmatrix}, \quad \begin{bmatrix} q_0 \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = \begin{bmatrix} 12.691 \\ -36.115 \\ 34.224 \\ -10.799 \end{bmatrix} \quad (6.30)$$

Measurements for 0.25 Hz frequency were made again with these new controller parameters and results shown in Fig. 6.32-Fig. 6.34 clearly prove the feed-forward design of the controller helped and the resulting controller experiences only a slight phase shift between reference and output values. It follows the reference value quite reliably and manages to make 8 complete revolutions in 35 seconds (as would be expected from 0.25 Hz signal).

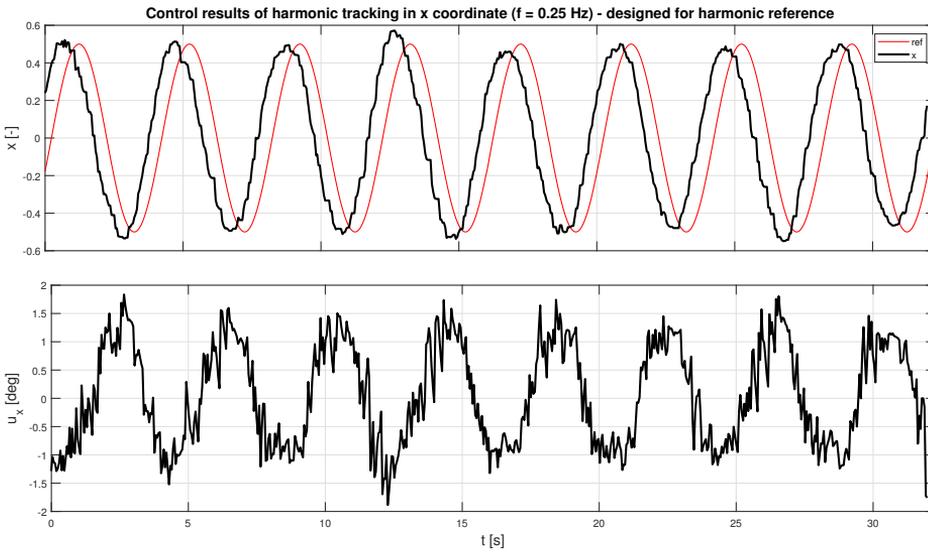


Fig. 6.32 Control results for harmonic tracking in x coordinate

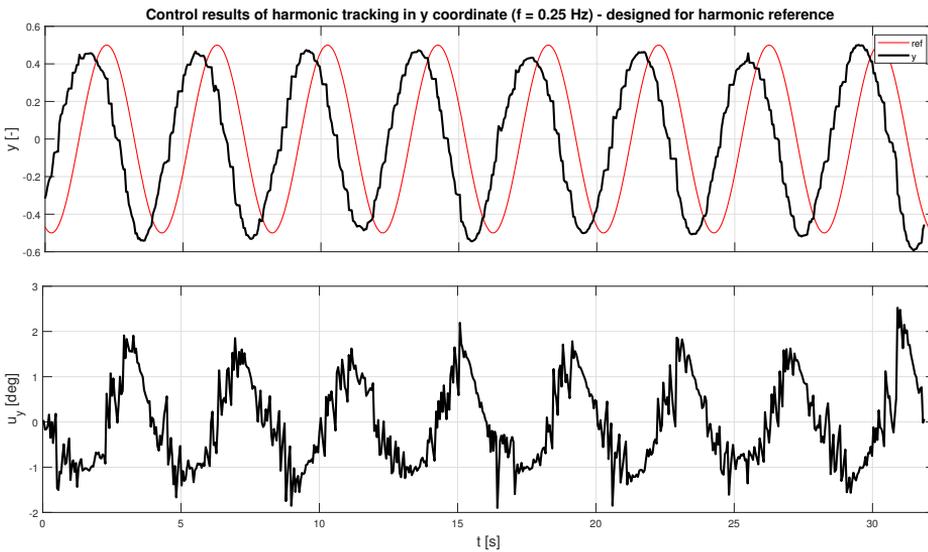


Fig. 6.33 Control results for harmonic tracking in y coordinate

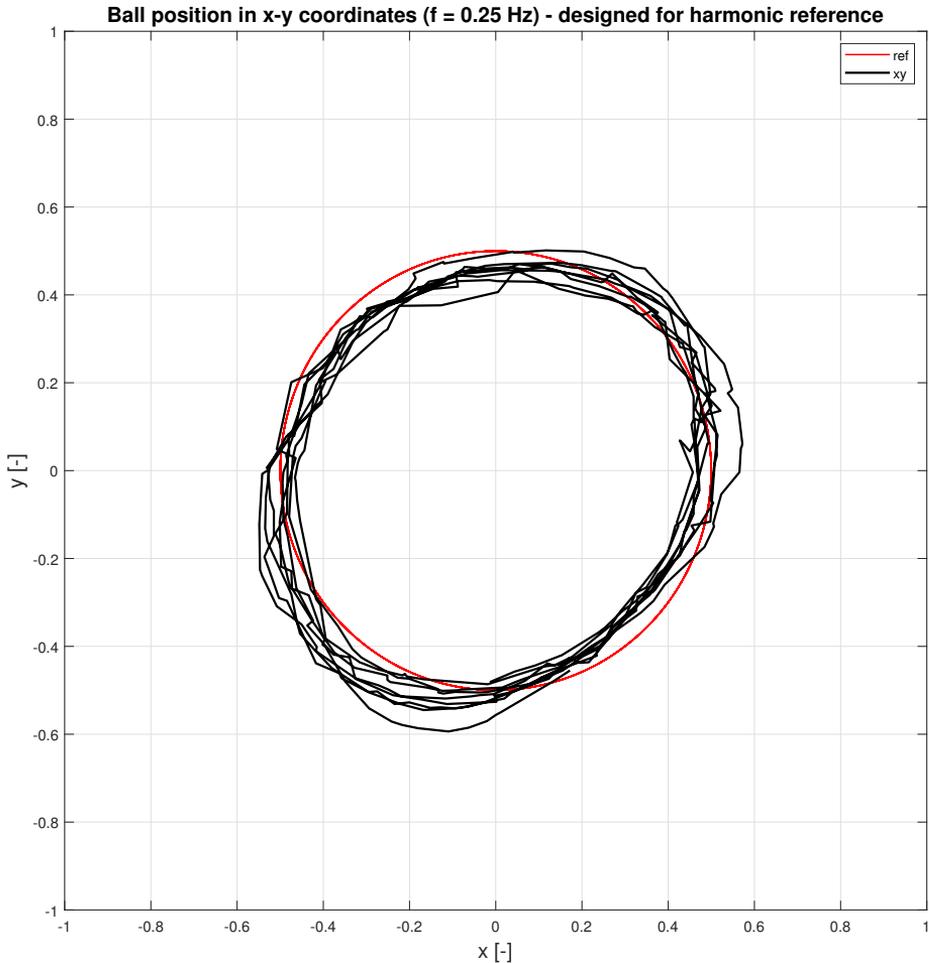


Fig. 6.34 Control results for harmonic tracking in the x-y plane

Quite interesting 3-D plot of the movement of the ball is shown in Fig. 6.35. It is similar to the x-y plot, but with time plotted on the z axis. It clearly shows the position of the ball in time-space coordinates (excluding height value z) and presents the result in another format. It shows deviations in time and provides a much better picture of motion of the ball in time. Previous 3 figures are actually just front, side and top views of this 3D plot. Its value sits in the more clearer picture of measured data and provides an extra view at the motion of the ball itself.

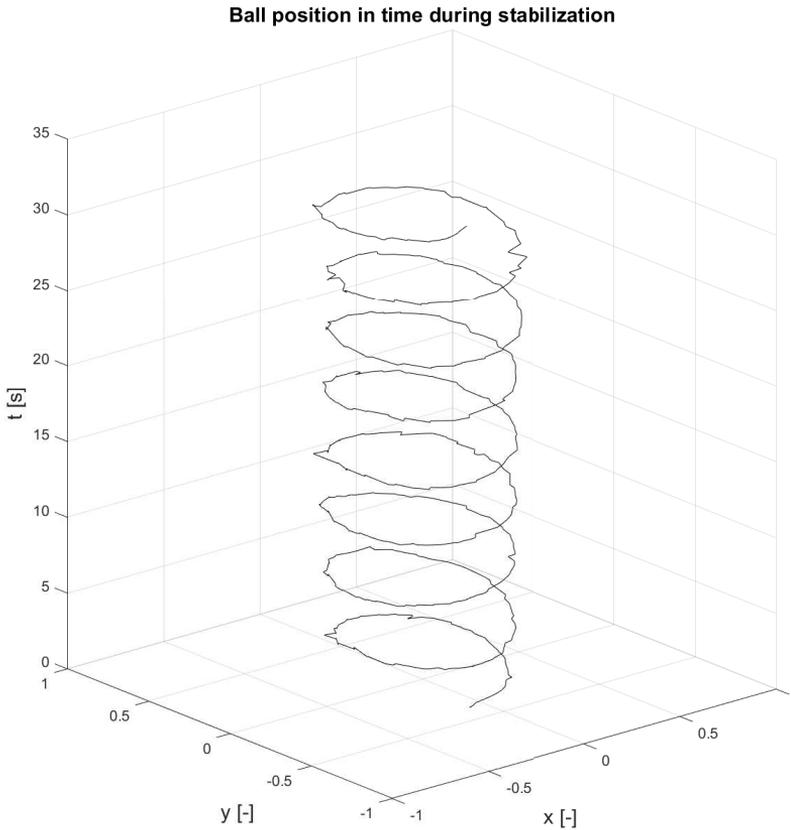


Fig. 6.35 Control results for harmonic tracking in the x-y-t space

7 CONTRIBUTION TO SCIENCE AND PRACTICE

The thesis explores the topic of control of the Ball & Plate model using a robotic manipulator with 7 degrees of freedom which can be further used for educational, research, or testing purposes. A strong emphasis is put on the resulting controller effort which needs to be bound to certain limitations of robotic manipulators concerning their prolonged and repetitive use for the same task. Their transmissions and gearboxes need to withstand the control actions of the algorithm to satisfy the long operating hours needed and expected during the life cycles of these ma-

nipulators. A great advantage of using a manipulator with 7 degrees of freedom is its versatility in exploiting also translational movements of the whole plate in space. This can be achieved also by a lower degree of freedom manipulator, but the 7th axis makes it possible to choose a different configuration of the manipulator (one of them is presented in Fig. 5.8) which greatly extends the kinematic flexibility of the proposed system. In addition, the introduction of errors into the system is much easier by exploiting the independent movement of one of the axes or by swiftly moving the whole plate in space. Education-oriented contribution is thus undeniable, especially in mechatronic study programs which probably already have a robotic manipulator present in their laboratories, so no other mechanical equipment is needed to have a ready-to-go Ball & Plate model. The model can be also used for testing various algorithms in different conditions and scenarios, quickly adapting to researchers' needs. Besides this it heavily relies on a real-world usage which is quite important and often over-looked aspect of many works. The practical usage and quick deployment on any type of robot available is and advantage suitable for better modularity of the whole system.

Peek in the Future

The trend for the following years shows much greater automation in fields that fall out of the industrial standards because of a lack of experienced staff. Robots are going to be used for many jobs that are currently thought of as hard to automate and with the need for the supervision of a human operator. Many of these tasks in the future may require some form of control of the handled process and having a simple approach to designing a suitable controller will be a necessity. This thesis shows such a controller can be set up quickly with just a few parameters for fine-tuning to achieve a desired behavior and that the whole process can be automated quite substantially. This paves the way to no-code/low-code integrations that require as little experience and skill as possible. The deployment of these solutions will be thus much faster and leaner, and no experts in the field will be required to complete a given task. These solutions will be most probably also directly connected to AI systems, data analysis, and machine vision to better utilize the technologies available.

Control of Unstable Systems Made Easy

The controller design strategy presented in this thesis is showing that a (semi-optimal) 2DoF polynomial controller design approach is quite a good fit for the control of unstable systems where it can provide performance, robustness, stabilization, disturbance rejection, and trajectory tracking on par with other methods and is a worthy competitor among the vast options of different controller design strategies found in the literature. It is able to easily stabilize the process without a deep knowledge of the system which is more than suitable for quick deployment. Designing a controller for fast unstable systems is a challenge because the testing without a properly designed controller results in very unstable behavior of controlled variables and unpredictable behavior of the actuation system. The controller thus needs to be able to stabilize the system before the quality of the control can be improved and fine-tuned on the real device. The controller described in this thesis can robustly stabilize the controlled plant making any fine-tuning of its parameters quicker and more user-friendly.

Service Life of Motion Systems During Control

Another limitation this thesis tackled was the service life of the robot's components while controlling a fast unstable process. Fast processes are directly linked to fast actuation and although industrial robots are designed for continuous operation with a relatively large mean time to failure, they are still prone to rapid (and unpredictable) changes in their movements. This can cause a lot of issues during their operation while controlling fast unstable systems. This thesis showed that the designed controller is able to stabilize the system at the same time as other common strategies, but with really low controller effort (especially for quick step-changes). There is always some compromise needed between quality, speed, reliability, robustness, and effort of the control and controller approach described in this thesis finds the optimal equilibrium between them. This feature, combined with ease of integration of the design, proves this is a competitive way on how to approach the problem of service life in these applications.

An Example

Bipedal locomotion is an unstable system and many robotic applications are aimed at this topic. It was proved that algorithms used in this thesis are more than able to stabilize the unstable system while keeping the load on actuators as low as possible and still maintaining comparable results as more standard forms of control. The algorithm calculates a semi-optimal solution to the problem, dealing with the unstable part and leaving fewer parameters to set, which are easily manageable and their effect on the whole system is more predictable in certain cases. The robustness of the proposed algorithms was not the main goal of this thesis, but it was shown that the controller can offer solid stabilization even with delayed and noisy communication between controller and actuator, although with much worse quality. All these criteria are important in real-world applications for continuous operation over multiple hours without failure and malfunctions. Algorithms proposed in this thesis can be easily made adaptive offering even more value.

8 CONCLUSION

This thesis discussed the theoretical context of the Ball & Plate problem and its solution utilizing a collaborative robotic manipulator as the electromechanical component of the model. It also described the design and utilization of a 2 DoF LQ polynomial controller for the specified problem and compared it in simulation with typical controller types implemented in B&P problems. The spectral factorization of polynomials was investigated in order to find a more optimal solution, compensating for the dynamics of the controlled system while maintaining the controller effort (and its rapid change) within the constraints of the manipulator. Before the experimental part was tested on the real manipulator, it was constructed in a simulation environment to check the proposed methods and approaches. This feasibility study was then implemented with confidence in the real system and rigorously evaluated for ball stability, disturbance rejection, and harmonic reference tracking. Education in automation courses largely

motivated by robots and industrial robotic manipulators can benefit from the findings. The path to the result extends through several disciplines and displays not only controller design ideas, but also controller-robot operation and communication, kinematics and dynamics of the robot, and the real application of the problem with its specific limits.

Ball & Plate model application on a collaborative 7-axis manipulator is not the optimal solution to this problem, but the B&P model is the best example of such a system that can be employed in laboratory conditions (together with inverted pendulum). Applications of bipedal robots can benefit tremendously from the methods suggested in this thesis, as these robots must navigate space while stabilizing their own bodies, battery packs, and, most crucially, random external forces acting on them. These applications have existed for a number of decades, but the movement of bipedal robots relied mostly on shifting the weight from one leg to another, thereby significantly lowering the instability of the movement itself. The strain on the actuators, gears, and other mechanical parts of these robots deployed in real-world applications is another crucial characteristic. Numerous controllers fail to maintain the optimal balance between rapid stabilization and low controller effort. In addition, reduced controller effort reduces the power consumption of the entire system, hence cutting operating expenses and extending the battery life of robots that require them. Thus, the results presented in this thesis conform to the outlined criteria and reliably compete with established control theory methods.

REFERENCES

- [1] C. An and C. Atkeson. *Model-based control of a robot manipulator*. MIT Press, Cambridge, Mass, 1988.
- [2] T. C. Hsia and L. S. Gao. Robot manipulator control using decentralized linear time-invariant time-delayed joint controllers. In *Proceedings., IEEE International Conference on Robotics and Automation*, 1990.
- [3] F. Lewis, D. Dawson, and C. Abdallah. *Robot manipulator control : theory and practice*. Marcel Dekker, New York, 2004.
- [4] R. Kelly, V. Santibanez, and A. Loria. *Control of robot manipulators in joint space*. Springer, London, 2005.
- [5] M. Khan, M. Ul Islam, and J. Iqbal. Control strategies for robotic manipulators. 2012.
- [6] J. Lee, P. H. Chang, and M. Jin. Adaptive integral sliding mode control with time-delay estimation for robot manipulators. *IEEE Transactions on Industrial Electronics*, 64, 2017.
- [7] O. Khatib and J. Burdick. Motion and force control of robot manipulators. In *Proceedings. 1986 IEEE International Conference on Robotics and Automation*, volume 3, 1986.
- [8] G. Zeng and A. Hemami. An overview of robot force control. *Robotica*, 15, 1997.
- [9] L. Basanez and J. Rosell. Robotic polishing systems. *Robotics & Automation Magazine, IEEE*, 12, 2005.
- [10] A. Perrusquia, W. Yu, and A. Soria. Position/force control of robot manipulators using reinforcement learning. *Industrial Robot: the international journal of robotics research and application*, 46, 2019.
- [11] A. Mishra and O. Meruvia-Pastor. Robot arm manipulation using depth-sensing cameras and inverse kinematics. *2014 Oceans - St. John's, OCEANS 2014*, 2015.

- [12] H. Behzadi-Khormouji, V. Derhami, and M. Rezaeian. Adaptive visual servoing control of robot manipulator for trajectory tracking tasks in 3d space. In *2017 5th RSI International Conference on Robotics and Mechatronics (ICRoM)*, 2017.
- [13] P. Neto, J. N. Pires, and A. Moreira. Accelerometer-based control of an industrial robotic arm. 2009.
- [14] P. Useche Murillo, R. Moreno, O. Avilés Sánchez, and S. Titular. Individual robotic arms manipulator control employing electromyographic signals acquired by myo armbands. *International Journal of Applied Engineering Research*, 11, 2016.
- [15] L. Gracia, J. Solanes, P. Muñoz-Benavent, J. Valls Miro, C. Perez-Vidal, and J. Tornero. Human-robot collaboration for surface treatment tasks. *Interaction Studies*, 20, 2019.
- [16] P. M. Kebria, A. Khosravi, S. Nahavandi, P. Shi, and R. Alizadehsani. Robust adaptive control scheme for teleoperation systems with delay and uncertainties. *IEEE Transactions on Cybernetics*, 50, 2020.
- [17] V. Girbes-Juan, V. Schettino, Y. Demiris, and J. Tornero. Haptic and visual feedback assistance for dual-arm robot teleoperation in surface conditioning tasks. *IEEE Transactions on Haptics*, 14, 2021.
- [18] C. González, J. E. Solanes, A. Muñoz, L. Gracia, V. Girbés-Juan, and J. Tornero. Advanced teleoperation and control system for industrial robots based on augmented virtuality and haptic feedback. *Journal of Manufacturing Systems*, 59, 2021.
- [19] K. A. Stone, S. Dittrich, M. Luna, O. Heidari, A. Perez-Gracia, and M. Schoen. Augmented reality interface for industrial robot controllers. In *2020 Intermountain Engineering, Technology and Computing (IETC)*, 2020.
- [20] T. Yoshikawa and A. Sudou. Dynamic hybrid position/force control of robot manipulators-on-line estimation of unknown constraint. *IEEE Transactions on Robotics and Automation*, 9, 1993.

-
- [21] M. Bjerkgeng, A. A. Transeth, K. Y. Pettersen, E. Kyrkjebø, and S. A. Fjerdingen. Active camera control with obstacle avoidance for remote operations with industrial manipulators: Implementation and experimental results. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2011.
- [22] A. Dzedzickis, J. Subačiūtė-Žemaitienė, E. Šutinys, U. Samukaitė-Bubnienė, and V. Bučinskas. Advanced applications of industrial robotics: New trends and possibilities. *Applied Sciences*, 12(1), 2022.
- [23] J. Bruce, C. Keeling, and R. Rodriguez. Four degree of freedom control system using a ball on a plate. Master's thesis, Southern Polytechnic State University, Georgia, US, 2011.
- [24] D. Stander, S. Jimenez-Leudo, and N. Quijano. Low-cost ball and plate design and implementation for learning control systems. In *2017 IEEE 3rd Colombian Conf. on Automatic Control (CCAC)*, 2017.
- [25] A. Jadlovska, S. Jajcisin, and R. Lonscak. Modelling and PID Control Design of Nonlinear Educational Model Ball & Plate. In *17th International Conference on Process Control '09*, Strbske Pleso, 2009.
- [26] S. Awatar, C. Bernard, N. Boklund, A. Master, D. Ueda, and K. Craig. Mechatronic design of a ball-on-plate balancing system. *Mechatronics*, 12, 2002.
- [27] A. Kassem, H. HADDAD, and C. Albitar. Comparison between different methods of control of ball and plate system with 6dof stewart platform. *IFAC-PapersOnLine*, 48, 2015.
- [28] A. Koszewnik, K. Troc, and M. Slowik. Pid controllers design applied to positioning of ball on the stewart platform. *Acta Mechanica et Automatica*, 8, 2014.
- [29] K. A. Khan, R. R. Konda, and J.-C. Ryu. Ros-based control for a robot manipulator with a demonstration of the ball-on-plate task. *Advances in Robotics Research*, 2, 2018.

- [30] K.-K. Lee, G. Batz, and D. Wollherr. Basketball robot: Ball-on-plate with pure haptic information. In *2008 IEEE International Conference on Robotics and Automation*, 2008.
- [31] C.-L. Shih, J.-H. Hsu, and C.-J. Chang. Visual feedback balance control of a robot manipulator and ball-beam system. *Journal of Computer and Communications*, 5, 2017.
- [32] H. Wang, Y. Tian, Z. Sui, X. Zhang, and C. Ding. Tracking control of ball and plate system with a double feedback loop structure. In *2007 International Conference on Mechatronics and Automation*, 2007.
- [33] M. Moarref, M. Saadat, and G. Vossoughi. Mechatronic design and position control of a novel ball and plate system. In *2008 16th Mediterranean Conference on Control and Automation*, 2008.
- [34] Y. Tian, M. Bai, and J. Su. A non-linear switching controller for ball and plate system. *International Journal of Modelling, Identification and Control*, 1, 2006.
- [35] A. Ghiasi and H. Jafari. Optimal robust controller design for the ball and plate system. In *9th International Conference on Electronics Computer and Computation (ICECCO 2012)*, Ankara, Turkey, 2012.
- [36] V. Bobal, J. Bohm, J. Fessler, and J. Machacek. *Digital Self-tuning Controllers*. Springer-Verlag London, 2005.
- [37] M. Sebek and V. Kucera. Polynomial approach to quadratic tracking in discrete linear systems. *IEEE Transactions on Automatic Control*, 27, 1982.
- [38] PolyX - The Polynomial Toolbox for use with MATLAB. <http://www.polyx.com/>. Accessed: 2022-06-07.
- [39] J. Fryman and B. Matthias. Safety of industrial robots: From conventional to collaborative applications. *7th German Conference on Robotics, ROBOTIK*, pages 1–5, 2012.
- [40] A. A. Malik and A. Bilberg. Collaborative robots in assembly: A practical approach for tasks distribution. *Procedia CIRP*, 81, 2019.

-
- [41] M. Knudsen and J. Kaivo-oja. Collaborative robots: Frontiers of current literature. *Journal of Intelligent Systems: Theory and Applications*, 3, 2020.
- [42] F. Vicentini. Collaborative robotics: A survey. *Journal of Mechanical Design*, 143, 2020.
- [43] ABB. *Product specification IRB 14000*, 2021. Rev. Q.
- [44] YuMi® - IRB 14000 | Collaborative Robot. <https://new.abb.com/products/robotics/collaborative-robots/irb-14000-yumi>. Accessed: 2022-03-13.
- [45] M. Taghbalout, J. F. Antoine, and G. Abba. Experimental dynamic identification of a yumi collaborative robot. *IFAC-PapersOnLine*, 52, 2019. 9th IFAC Conference on Manufacturing Modelling, Management and Control MIM 2019.
- [46] M.-T. Ho, Y. Rizal, and L.-M. Chu. Visual servoing tracking control of a ball and plate system: Design, implementation and experimental validation. *International Journal of Advanced Robotic Systems*, 10(7), 2013.
- [47] F. C. Braescu, L. Ferariu, R. Gilca, and V. Bordianu. Ball on plate balancing system for multi-discipline educational purposes. In *2012 16th International Conference on System Theory, Control and Computing (ICSTCC)*, 2012.
- [48] C. Connolly. Technology and applications of abb robotstudio. *Industrial Robot: An International Journal*, 36, 2009.
- [49] Z. Liqiu, A. Juan, Z. Ronghao, and M. Hairong. Trajectory planning and simulation of industrial robot based on matlab and robotstudio. 2021.
- [50] F. Caputo, M. Caterino, A. Luca, M. Fera, A. Greco, G. Lamanna, R. Macchiaroli, P. Manco, M. Manzo, and D. Perfetto. Product and process integrated design to enhance smart manufacturing systems. 2019.
- [51] P. Naslin. *Essentials of optimal control*. Iliffe, 1968.

Publications of the Author

- [P.1] SPACEK, L., BOBAL, V. and VOJTESEK, J. Digital Control of Ball & Plate Model Using LQ Controller, In *Proceedings of the 2017 21st International Conference on Process Control (PC)*, 36-41. Editors: Fikar, M. and Kvasnica, M., Slovakia 2017.
- [P.2] SPACEK, L., BOBAL, V. and VOJTESEK, J. LQ Digital Control of Ball & Plate System, In *Proceedings of 31st European Conference on Modelling and Simulation ECMS 2017*, 427-432. Hungary 2017.
- [P.3] SPACEK, L., VOJTESEK, J. and BOBAL, V. Educational Model of Unstable MIMO System, In *Proceedings of the 2017 23rd International Conference on Engineering*, 440-445. Technology and Innovation (ICE/ITMC), Portugal 2017.
- [P.4] SPACEK, L., BOBAL, V. and VOJTESEK, J. Maze Navigation on Ball & Plate Model, In *Cybernetics and Mathematics Applications in Intelligent Systems*, Advances in Intelligent Systems and Computing, vol. 574, Springer 2017.
- [P.5] SPACEK, L. and VOJTESEK, J. Overview of Ball & Plate Application for Collaborative Robot YuMi, In *Lecture Notes in Electrical Engineering*, vol. 505, Portugal 2019.
- [P.6] SPACEK, L., VOJTESEK, J. and ZATOPEK, J. Collaborative Robot YuMi in Ball and Plate Control Application: Pilot Study, In *Cybernetics and Algorithms in Intelligent Systems*, Advances in Intelligent Systems and Computing, vol. 765, Springer 2018.
- [P.7] SPACEK, L., VOJTESEK, J., GAZDOS, F. and KADAVY, T. Ball & Plate Model for Robotic System, In *Proceedings of 32nd European Conference on Modelling and Simulation ECMS 2018*, 226-231. Germany 2018.
- [P.8] SPACEK, L. and VOJTESEK, J. Ball & Plate Model on ABB YuMi Robot, In *Cybernetics and Automation Control Theory Methods in Intelligent*

- Algorithms*, Advances in Intelligent Systems and Computing, vol. 986, Springer 2019.
- [P.9] SPACEK, L. and VOJTESEK Implementation of 7 DoF Robotic System for Fast Unstable Processes, In *Proceedings of 33rd European Conference on Modelling and Simulation ECMS 2019*, Italy 2019.
- [P.10] SPACEK, L., VOJTESEK, J. and GAZDOS, F. Control of Unstable Systems Using a 7 DoF Robotic Manipulator, In *Machines 2022*, 10, 2022.
- [P.11] VOJTESEK, J., SPACEK, L. and DOSTAL, P. Simulation Study of 1DOF Hybrid Adaptive Control Applied on Isothermal Continuous Stirred-Tank Reactor, In *Proceedings of 31st European Conference on Modelling and Simulation ECMS 2017*, 446-452. Hungary 2017.
- [P.12] BOBAL, V., SPACEK, L. and HORNAK, P. Verification of Robust Properties of Digital Control Closed-Loop Systems, In *Proceedings of 31st European Conference on Modelling and Simulation ECMS 2017*, 348-354. Hungary 2017.
- [P.13] VOJTESEK, J., SPACEK, L. and ZATOPEK, J. MATLAB as a Tool for Modelling and Simulation of the Nonlinear System, In *Cybernetics and Algorithms in Intelligent Systems*, Advances in Intelligent Systems and Computing, vol. 765, Springer 2018.
- [P.14] VOJTESEK, J. and SPACEK, L. Adaptive Control of Temperature Inside Plug-Flow Chemical Reactor Using 2DOF Controller, In *Lecture Notes in Electrical Engineering, vol. 505*, Portugal 2019.
- [P.15] VOJTESEK, J., SPACEK, L. and GAZDOS, F. Control Of Temperature Inside Plug-Flow Tubular Chemical Reactor Using 1DOF And 2DOF Adaptive Controllers, In *Proceedings of 32nd European Conference on Modelling and Simulation ECMS 2018*, 239-245. European Council for Modelling and Simulation, Germany 2018.

CURRICULUM VITAE

**Lubos
Spacek**

Nationality: Slovak

CONTACT

 Drietoma 225,
91303 Drietoma, Slovakia
(Home)

 lu.spacek@gmail.com

 lspacek@utb.cz

 (+420) 720498690



europass

WORK EXPERIENCE

01/09/2019 – CURRENT – Prague, Czechia

Application engineer

ABB s.r.o., Robotics & Discrete Automation Division

01/06/2017 – 31/08/2019 – Prague, Czechia

Marketing specialist

ABB s.r.o., Robotics & Discrete Automation Division
Propagation and programming of marketing applications for fairs

01/06/2016 – 31/08/2019 – Zlín, Czechia

Automation engineer

Kovarna VIVA a.s.

01/02/2012 – 31/05/2016 – Zlín, Czechia

University research assistant

Tomas Bata University in Zlín

01/07/2015 – 31/08/2015 – Nova Dubnica, Slovakia

Engineering assistant

EVPU – Electrotechnical research and projecting company
Internship

01/07/2014 – 15/09/2014 – Roznov pod Radhostem, Czechia

Embedded systems software developer

Fresscale Semiconductor (currently NXP)
Internship

EDUCATION AND TRAINING

2011 – 2014 – Czechia

BEng

Tomas Bata University in Zlín

Address Czechia

2014 – 2016 – Czechia

MEng

Tomas Bata University in Zlín

Address Czechia

2016 – CURRENT – Czechia

PhD

Tomas Bata University in Zlín

Address Czechia

03/2016 – 04/2016

- Online course - Data Science and Machine Learning Essentials (Azure ML, Python, R)

edX.org

01/2016 – 02/2016

- Online course - Control System Design

edX.org - MIT

09/2015 – 12/2015

- Online course - Introduction to Computer Science and Programming Using Python

edX.org - MIT

LANGUAGE SKILLS

MOTHER TONGUE(S): Slovak

OTHER LANGUAGE(S):

English

Listening	Reading	Spoken production	Spoken interaction	Writing
C1	C1	C1	B2	B2
Russian				
A2	B1	A1	A1	A1

DIGITAL SKILLS

My Digital Skills

Microsoft Office / RobotStudio / C# / Python / C/C++ / JavaScript / React / SQL / TIA Portal / Siemens NX / Linux / Git / NoSQL / REST / Matlab/Simulink