



VYSOKÉ UČENÍ TECHNICKÉ V BRNĚ
BRNO UNIVERSITY OF TECHNOLOGY

FAKULTA STROJNÍHO INŽENÝRSTVÍ
FACULTY OF MECHANICAL ENGINEERING

ÚSTAV AUTOMATIZACE A INFORMATIKY
INSTITUTE OF AUTOMATION AND COMPUTER SCIENCE

**MODEL BALL & PLATE: SIMULACE A
NÁVRH ŘÍZENÍ**
BALL & PLATE MODEL: SIMULATION AND CONTROL DESIGN

DIPLOMOVÁ PRÁCE
MASTER'S THESIS

AUTOR PRÁCE
AUTHOR

Bc. Sofiia Burlachenko

VEDOUCÍ PRÁCE
SUPERVISOR

doc. Ing. Radomil Matoušek, Ph.D.

BRNO 2019

Zadání diplomové práce

Ústav:	Ústav automatizace a informatiky
Studentka:	Bc. Sofiia Burlachenko
Studijní program:	Strojní inženýrství
Studijní obor:	Aplikovaná informatika a řízení
Vedoucí práce:	doc. Ing. Radomil Matoušek, Ph.D.
Akademický rok:	2018/19

Ředitel ústavu Vám v souladu se zákonem č.111/1998 o vysokých školách a se Studijním a zkušebním řádem VUT v Brně určuje následující téma diplomové práce:

Model Ball & Plate: simulace a návrh řízení

Stručná charakteristika problematiky úkolu:

Pro již existující laboratorní model Ball & Plate verifikujete simulační model a navrhnete algoritmus řízení.

Cíle diplomové práce:

Popis simulačního modelu soustavy Ball & Plate a její identifikace.

Simulační modelování vedoucí k návrhu regulátoru (geometrické místo kořenu, stavový regulátor).

Verifikace navrženého algoritmu řízení pomocí SimScape a reálného modelu.

Seznam doporučené literatury:

KATSUHIKO, Ogata: Modern Control Engineering (5th Edition), Pearson, ISBN-13: (2009) 978-0136156734

Termín odevzdání diplomové práce je stanoven časovým plánem akademického roku 2018/19

V Brně, dne

L. S.

doc. Ing. Radomil Matoušek, Ph.D.
ředitel ústavu

doc. Ing. Jaroslav Katolický, Ph.D.
děkan fakulty

ABSTRAKT

Tato práce se zabývá identifikací a regulací modelu „Ball & Plate“. Práce obsahuje popis již existujícího reálného modelu a příslušný matematický a simulační model. K výpočtu regulátoru je použita metoda kořenového hodografu a metoda stavového prostoru, zejména zpětnovazební regulátor s integrátorem. Závěrečná část práce je věnovaná konstrukci 3D modelu pomocí Simulink a SimScape, který dobře popisuje a vizualizuje chování reálného modelu a umožňuje snadno a rychle provádět simulační experimenty.

ABSTRACT

This thesis deals with the identification and regulation of the "Ball & Plate" model. The thesis contains a description of the existing real model and the relevant mathematical and simulation model. The root hodograph method and the state space method are used to calculate the controller, especially the feedback controller with integrator. The final part of the work is devoted to the 3D model construction using Simulink and SimScape, which describes and visualizes the behavior of the real model and enables simulation experiments to be performed quickly and easily.

KLÍČOVÁ SLOVA

Model kulička na ploše, CE 151, identifikace, řízení, kořenový hodograf, stavový prostor, Matlab, Simulink

KEYWORDS

Ball & Plate model, CE 151, identification, control, root locus, state space, Matlab, Simulink

BIBLIOGRAFICKÁ CITACE

BURLACHENKO, S ofiia. *Model Ball & Plate: simulace a návrh řízení*, Brno, 2019.
Diplomová práce. Vysoké učení technické v Brně, Fakulta strojního inženýrství,
Ústav automatizace a informatiky. Vedoucí práce doc. Ing. Radomil Matoušek, Ph.D.

PODĚKOVÁNÍ

Ráda bych na tomto místě poděkovala vedoucímu své diplomové práce doc. Ing. Radomilu Matouškovi, Ph.D. a Bc. Zdeňku Cejkovi za průběžnou pomoc a odborné konzultace.

ČESTNÉ PROHLÁŠENÍ

Prohlašuji, že tato práce je mým původním dílem, zpracovala jsem ji samostatně pod vedením doc. Ing. Radomila Matouška, Ph.D. a s použitím literatury uvedené v seznamu literatury.

V Brně dne 1. 3. 2019

.....

Sofia Burlachenko

OBSAH

1	ÚVOD.....	15
2	TEORETICKÁ ČÁST.....	17
2.1	Diskuze k modelu a rešerše literatury.....	17
2.2	Model “Ball & Plate”.....	18
2.2.1	Popis modelu.....	18
2.2.2	Parametry modelu.....	19
2.2.3	Programová podpora.....	20
2.2.4	Matematický model.....	22
2.2.5	Linerizace a zjednodušení modelu.....	25
2.2.6	Senzor polohy.....	25
2.2.7	Servosystém modelu.....	26
2.3	Stavové řízení.....	28
2.3.1	Popis systému ve stavovém prostoru.....	28
2.3.2	Stavový popis a přenos systému.....	29
2.3.3	Řiditelnost a pozorovatelnost systému.....	30
2.3.4	Stavový zpětnovazební regulátor s integrátorem.....	31
2.4	Metoda geometrického místa kořenů.....	32
2.4.1	Teoretické poznámky.....	32
2.4.2	Vlastnosti metody geometrického místa kořenů.....	33
3	PRAKTICKÁ ČÁST.....	35
3.1	Seznámení s modelem.....	35
3.2	Identifikace.....	38
3.3	Návrh regulátoru.....	43
3.3.1	Metoda geometrického místa kořenu.....	43
3.3.2	Metoda stavového prostoru.....	47
3.4	Výsledky experimentu na reálném modelu.....	49
3.5	Počítačová simulace.....	50
3.5.1	Konstrukce 3D modelu.....	50
3.5.2	Identifikace a regulace modelu.....	51
3.5.3	Výsledky simulací 3D modelu.....	53
4	ZÁVĚR.....	57
	SEZNAM POUŽITÉ LITERATURY.....	59
	SEZNAM OBRÁZKŮ.....	61
	SEZNAM TABULEK.....	63
	SEZNAM ZKRATEK.....	65
	PŘÍLOHY.....	67

1 ÚVOD

V současném vzdělávacím systému je kladen velký důraz na rozvoj praktických dovedností studentů. Moderní přístup k výuce v laboratořích předpokládá využití skutečných (reálných, fyzikálních) modelů a to jak pro jejich identifikaci, tak i pro návrh řízení. Studenti provádějící laboratorní práci vykonávají následující hlavní kroky návrhu: vytvoření matematického modelu řízeného objektu (řízené soustavy), vývoj algoritmu pro její řízení, kontrola v simulačním prostředí, implementace řídicího algoritmu v reálném čase a finální ladění algoritmu regulace. Tímto způsobem je dosaženo přímé interakce studentů s reálným předmětem řízení, což je důležité při přípravě kvalifikovaných odborníků v oblasti systémů automatického řízení.

Tato práce se zabývá konkrétním modelem označovaným jako *“Ball and Plate”* od společnosti Humusoft, konkrétněji jeho popisem, simulačním modelováním a návrhem řízení. Uvedený model představuje nestabilní systém s astatismem druhého řádu, který má dva vstupy a jeden, resp. dva říditelné výstupy. Řízení probíhá v reálném čase a jako zpětnou vazbu k určení polohy míče na desce využívá počítačového vidění. Hlavním zadáním úlohy je ovládat kuličku na desce - řídit její polohu dle zadaných souřadnic. Svým pojetím poskytuje model *“Ball & Plate”*, dále B&P, prostor pro široké spektrum experimentů v oblasti automatického řízení.

Diplomová práce popisuje vlastní simulační model, zahrnuje jeho identifikaci a návrh vhodného regulátoru schopného řídit systém v požadované kvalitě. Pro návrh regulátoru byla použita metoda kořenového hodografu a metoda stavového prostoru. Popis těchto metod lze najít v teoretické části práce. Navržené regulátory byly verifikovány simulačně a následně testovány na reálném modelu. Součástí řešení je 3D schéma modelu sestavené s využitím knihovny SimScape-Simmechanics v prostředí Simulink. Schéma se skládá ze dvou nezávislých systémů realizujících pohyb kuličky ve dvou nezávislých osách, přičemž každá z nich je regulována samostatně. Pro potřeby identifikace soustavy je použita metoda nejmenších čtverců. Navržená regulace je založena na zpětnovazebním regulátoru s integrátorem na vstupu. Výsledný pohyb kuličky představuje superpozice pohybu kuliček v obou nezávislých osách.

Reálným smyslem této práce bylo seznámit se s uvedeným modelem, navrhnout řídicí algoritmus, který zohlední specifika systému a získat praktické dovednosti ve zvoleném oboru – automatizace.

2 TEORETICKÁ ČÁST

2.1 Diskuze k modelu a rešerše literatury

Model „*Ball & Plate*“ (B&P) je poměrně populární systém pro automatické řízení. Na toto téma existuje spousta závěrečných prací a článků. Tato část obsahuje stručný popis některých z nich, které byly dostupné.

Ing. Miroslav Kirchner [1] ve své diplomové práci s tímto modelem použil pro ro identifikaci řešič v Excelu, pro video analýzu program *Viana3*, pro výpočet regulátoru použil metodu geometrického místa kořenu a Naslinovou metodu. Podle autora se Naslinova metoda ukázala jako neefektivní, regulovala s vysokým překmitem a dlouhou dobou ustálení. Pro metodu kořenového hodografu dosáhl PD regulátor nejlepší výsledky, v reálném nasazení však tento regulátor zcela selhal a nebyl schopen dosáhnout žádané polohy.

Ing. Luboš Spaček [2] provedl identifikaci modelu se dvěma různými kuličkami. Zabýval se výpočtem LQ regulátoru a naznačil, že řešení je dostatečně spolehlivé, aby se dokázalo vyrovnat i s přepínáním míče „v letu“. Aby byl proces výběru referenčního signálu zajímavější a měl nějaký algoritmus, byl implementován průchod bludištěm. Pro účely modelování byla autorem vyvinuta GUI aplikace, která se ukázala být cenným nástrojem pro rychlou změnu referenčních hodnot a zobrazení konečného výsledku řízení reálného modelu.

Bc. Michal Debre [3] pro identifikaci používal metodu *Grey-Box*, ale kvůli nedostatku dat musel udělat model, který bude popisovat oba motory současně, což úspěšně zvládl. PID regulátor navrhl empirickou metodou, která se ukázala jako nejvhodnější. Navrhnutý LQ regulátor na skutečném systému nefungoval z důvodu působení externích poruch. Výsledkem jeho práce je teda ucelený model, který svým popisem s využitím pokročilých metod identifikace velmi přesně popisuje reálný systém, dva spočítané regulátory, které byly otestované na reálném modelu.

Ing. Jitka Míková [4] se také používala stavové řízení, konkrétně metodu přiřazení pólů a LQ řízení. Za tímto účelem aplikovala Luenbergerův rekonstruktor stavu. Dále využívala polynomiální metody pro dvě základní konfigurace řízení – 1DoF a 2DoF. Nejlepších výsledků dosáhli regulátory typu I navržené polynomiálními metodami 1DoF a 2DoF, ale za cenu vysokých překmitů. Naopak nejhorších výsledků bylo dosaženo použitím stavových regulátorů navržených metodou přiřazení pólů. Také byla vytvořena GUI aplikace v programu MATLAB, pro jednoduché a přehledné experimenty s modelem a to jak simulačně, tak v reálném čase

Článek „*Trajectory planning and tracking of ball and plate system using hierarchical fuzzy control scheme*“ od Xingzhe Fan, Naiyao Zhang, Shujie Teng navrhl [5] popisuje hierarchické schéma fuzzy řízení pro řešení problému plánování a sledování trajektorie systému „*Ball & Plate*“. Výsledky simulace ukázaly, že systém může ovládat míč z výchozího bodu do koncového bodu, aniž by zasáhl překážky.

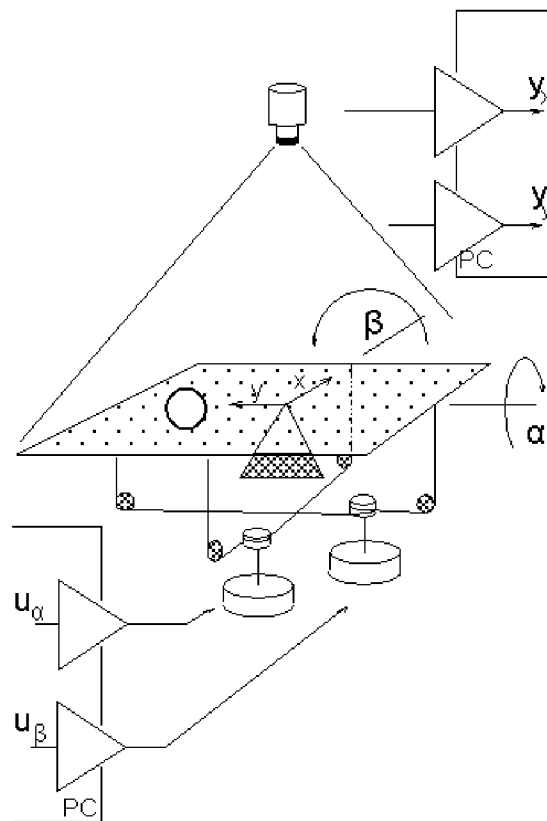
V článku *“Modelling of Ball and Plate System Based on First Principle Model and Optimal Control”* autoru František Dušek, Daniel Honc, Rahul Sharma K. [6] je matematický model systému získán Newton-Eulerovou metodou, která zohledňuje rovnováhu sil a momentů koule a desky, motorů a převodů. Nelineární model je linearizován kolem pracovních bodů. Simulační modelování pro ověření modelu bylo provedeno v otevřené smyčce. Linearizovaný model je zde využit pro diskrétní optimální řízení LQ v úloze sledování trajektorie. Výsledek simulace dokazuje kvalitu linearizace nelineárního modelu.

Článek *“Optimal control of the mechatronical laboratory model B&P_KYB”* autoru Matej Oravec a Anna Jadlovská [7] prezentuje laboratorní model B&P z hlediska konstrukčního návrhu, procesu vytváření simulačního modelu na základě fyzikálních zákonů a návrhu optimálního řízení modelu. Simulační model byl vytvořen ve formě *Grey-Box* modelu. Ukázalo se, že tento model je vhodnou náhradou skutečného modelu a může být použit pro následné simulační modelování. Výsledky simulace LQ s integrátorem v diskrétní formě neposkytly přijatelné výsledky, protože odezva reálného nebo simulačního modelu B&P na změnu referenční trajektorie byla zpožděna. Spojení mezi skutečným modelem a PC pomocí webové kamery a jednočipového mikropočítače se ukázalo jako velmi efektivní.

2.2 Model “Ball & Plate”

2.2.1 Popis modelu

Model *“Ball & Plate”* (B&P) od firmy Humusoft je vzdělávací nástroj, resp. reálná model určený pro studenty technických inženýrských oborů. Jedná se o nestabilní systém s astatismem druhého řádu se dvěma vstupy a dvěma výstupy, který umožňuje experimenty s dynamickým modelováním, identifikací, analýzou a řízením pomocí číslicových regulátorů. Základní úlohou je ovládní trajektorie v osách x , y , volně pohyblivé koule umístěné na desce. Omezením úlohy jsou hrany (okraje) desky. Změna polohy koule je dosažena nakloněním desky podél os x a y , tj. změnou sklonu desky řízené dvěma krokovými motory. Obě osy lze ovládat nezávisle, přičemž jejich vzájemné ovlivňování je z důvodu nízké rychlosti a akcelerace míče zanedbatelné. Polohu kuličky sleduje kamera. Obraz z kamery je zpracováván počítačem a výsledkem je aktuální pozice kuličky na desce v osách x , y . Na základě informací o aktuální pozici kuličky pošle počítač příslušný signál k motoru, který nakloní desku na úhel, který zajistí přemístění kuličky do požadované polohy.



Obr. 1: Schéma modelu B&P (Ball & Plate) [8]

Diskutovaný reálný model (pozn.: označení výrobce CE 151 kulička na ploše) je navržen pro řízení v reálném čase prostřednictvím počítače, resp. pomocí I/O hardwarových prostředků počítače (např. MF624). Výhodou tohoto „studentsky přívětivého“ systému je jeho komplexnost, tj. není nutné využívat a připojovat další přídatná zařízení laboratoře. Veškeré nezbytné měření provádí software, student nemá přímý přístup k signálům přicházejícím z modelu a řídicím signálům jdoucím do modelu. Tedy neexistuje nebezpečí fyzického poškození systému studentem během měření.

2.2.2 Parametry modelu

Model se skládá z hlavní části, kterou je sklápěcí deska se servopohonem, USB kamery, multifunkční karty MF624 a zdroje napájení, jak je uvedeno v následujících tabulkách.

Hlavní část

tělo modelu	430 mm x 430 mm x 200 mm
naklápací deska	400 mm x 400 mm
tyč	1100 mm
hmotnost	9 kg
ovládání desky	2 krokové motory v otevřené smyčce
krok / úhel	1 krok $\approx 0,001^\circ$

Zdroj napájení

napájení	240V / 50Hz / 250mA nebo 110V / 60 Hz / 500mA
rozměr	175 x 175 x 100mm
váha	2 kg

Kamera

typ	Logitech QuickCam Pro 5000
rozměry	240 x 240 x 90mm
hmotnost	700 gm
maximální optické rozlišení	640 px x 480 px
pracovní rozlišení	160 px x 120 px
formát videa	RGB24
frekvence snímků	30 fps
vzdálenost od desky	500 mm

Multifunkční I / O karta

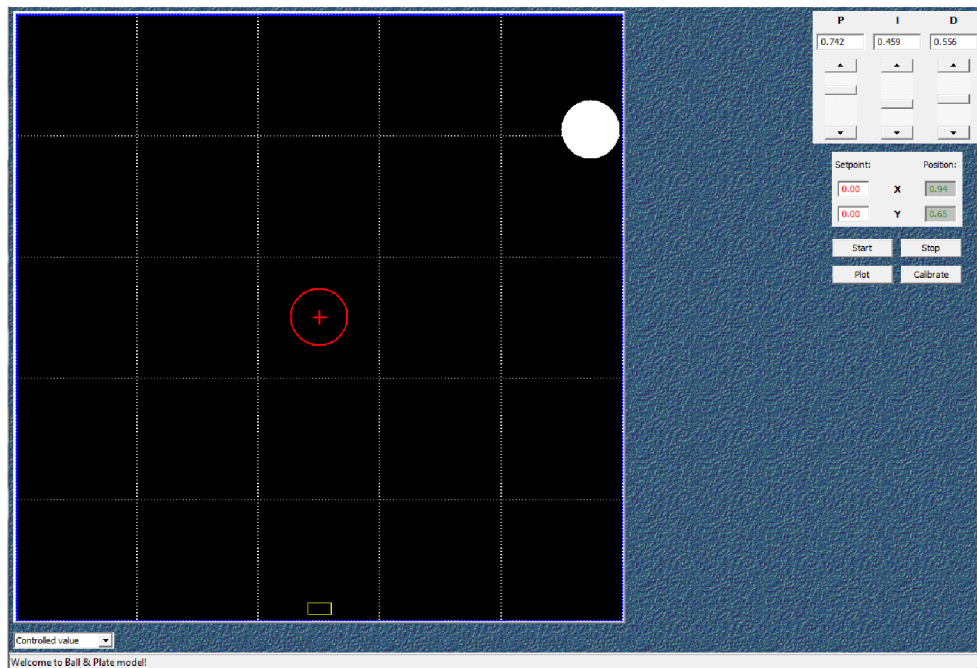
typ	MF 624 – PCI
doba převodu	4 ms
převodník A / D	8 14-bitových kanálů s jedním výstupem
převodník D / A	4 14-bitové kanály
digitální I / O	8 vstupů TTL a 8 výstupů TTL
další	4 vstupy enkoderu, 4 čítače / časovače

Rozsah možných experimentů:

- zpracování v reálném čase,
- návrh digitálního PID regulátoru,
- návrh regulátoru LQ / LQG,
- návrh fuzzy regulátoru,
- adaptivní návrh regulátoru,
- plánování cesty.

2.2.3 Programová podpora

Model B&P je dodáván s demo programem. Je napsán v jazyce C a je plně řízený pomocí menu program pro Windows. Umožňuje rychlé a pohodlné seznámení s tímto modelem. Program umožňuje sledovat pohyb míče, měnit parametry kamery, nastavit požadovanou a konečnou polohu míče, nastavit koeficienty PID regulátoru a provádět rychlé experimenty. Přátelské rozhraní s použitelnou grafikou usnadňuje seznámení s tímto modelem a sledování jeho chování v reálném čase.



Obr. 2: GUI dodávaného demo programu pro model B&P

Dalším dodávaným programem je program v Matlabu/Simulinku. Matlab je programovací jazyk vysoké úrovně a interaktivní prostředí pro programování, numerické výpočty a vizualizaci dat a výsledků. Pomocí Matlabu můžeme analyzovat data, vyvíjet algoritmy, vytvářet modely a programovat aplikace.

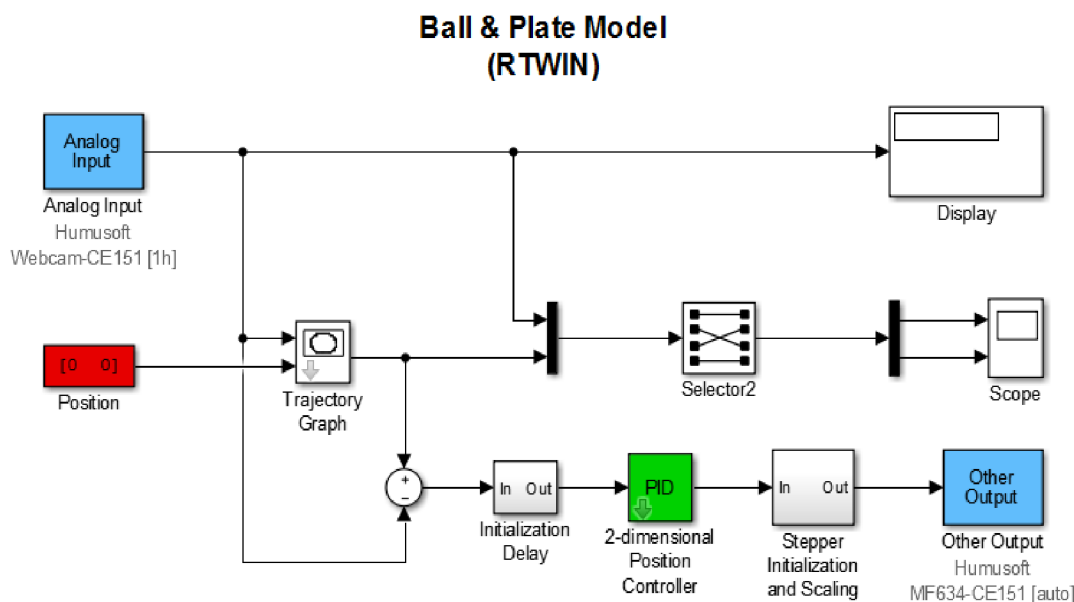
Simulink je grafické simulační prostředí, které umožňuje vytvářet dynamické modely pomocí blokových diagramů ve formě orientovaných grafů, včetně diskretních, spojitých a hybridních systémů. Jde o jeden z nepsaných standardů pro simulační modelování. Dodaný program v Simulinku nabízí počáteční diagram pro práci se systémem. Může být použit pro experimenty v reálném čase na simulačním i reálném modelu. Pro práci s tímto modelem jsou vyžadovány další toolboxy, jako je například Extended Real Time Toolbox, Control System Toolbox, Optimization Toolbox, System Identification Toolbox, Image Acquisition Toolbox.

Původní schéma v Simulinku obsahuje bloky:

- „Analog Input“, který je zodpovědný za čtení signálu z kamery a jeho další zpracování,
- „Display“ zobrazuje x a y souřadnice kuličky v rozsahu $[-1;1]$,
- „Position“ umožňuje nastavit požadovanou konečnou polohu kuličky,
- „Trajectory Graph“ obsahuje kód `'ce151traj'` pro kreslení pohybu kuličky, kde je výsledkem graf ukazující nastavenou koncovou polohu ve formě červeného kruhu a křivky pohybu koule na desce,
- „PID“ – dvoudimenzionální PID regulátor, umožňuje nastavit koeficienty P, I a D,
- „Other Output“ přenáší řídicí signál do motorů,

- „Scope“ ukazuje dva grafy se změnou polohy koule v souřadnicích x a y,
- „Stepper Initialization and Scaling“ pro krokovou inicializaci a škálování signálů do motorů,
- „Initialization Delay“ inicializuje zpoždění.

Původní schéma je znázorněno na obrázku 3:



Obr. 3: Základní Simulink/Matlab model B&P (pro CE 151)

2.2.4 Matematický model

Dynamika systému může být rozdělena na dvě části. Servo systém pro náklon desky a kuličku s volným pohybem na desce. Neexistuje žádná zpětná vazba od polohy kuličky k úhlu desky. Točivý moment generovaný kuličkou – když je blízko hrany desky, je mnohem menší než krouticí moment krokového motoru. Proto krokové motory včetně řídicí elektroniky jsou analyzovány odděleně od zbytku systému.[9]

Variační modelovací přístup je použit pro odvození obecného modelu dynamiky systému bez servosystému pro sklon desky, který bude modelován samostatně. Uvedme obecnou formu Euler-Lagrangeových rovnic

$$\frac{d}{dt} \frac{\partial W}{\partial \dot{q}_i} - \frac{\partial W}{\partial q_i} + \frac{\partial V}{\partial q_i} = Q_i. \quad (1)$$

kde proměnné mají následující význam:

- W celková kinetická energie,
- V celková potenciální energie,
- Q_i je i -ta obecná síla,
- q_i je i -ta obecná souřadnice,
- \dot{q}_i první derivace i -té obecné souřadnice podle času.

Systém má čtyři stupně volnosti, dva v pohybu kuličky na desce a dva v sklonu desky. Jako zobecněné souřadnice jsou zvoleny pozice x a y kuličky spojené s deskou a úhly α a β sklonu desky. Předpokládejme, že sklon desky je poháněn zobecněnými momenty τ_α a τ_β působícími na desku v odpovídajícím směru. Pro obecné shrnutí jsou zobecněným souřadnicím dle (1) přiřazeny proměnné q_i .

$$q_1 = x, \quad q_2 = y, \quad q_3 = \alpha, \quad q_4 = \beta \quad (2)$$

Pro usnadnění matematického modelování byly zavedeny následující předpoklady:

- mezi kuličkou a deskou není skluz,
- kontakt mezi kuličkou a deskou se nikdy neztrácí,
- neexistuje žádné tření (např. při kontaktu se vzduchem nebo mezi kuličkou a deskou),
- kulička je ideální koule a je homogenní,
- deska je nekonečná rovina a její sklon nemá žádnou hranici.

Pro další výpočty byly zvoleny následující proměnné a parametry:

x, y	souřadnice kuličky od středu desky [m],
r	poloměr kuličky [m],
m	hmotnost kuličky [kg],
I_b	moment setrvačnosti kuličky [kgm ²],
I_p	moment setrvačnosti desky [kgm ²],
α, β	úhly náklonu desky [rad],
ω	vektor úhlové rychlosti rotující kuličky [rad/s],
v	vektor rychlosti [m/s].

Kinetická energie systému se skládá z kinetické energie kuličky a kinetické energie desky:

$$W = W_p + W_b \quad (3)$$

Kinetická energie kuličky se skládá z translačních a rotačních částí:

$$W_b = W_r + W_t \quad (4)$$

Translační energie míče je:

$$W_t = \frac{1}{2} m v^2 = \frac{1}{2} m (\dot{x}^2 + \dot{y}^2) \quad (5)$$

Rotační energie míče:

$$W_r = \frac{1}{2} I_b \omega^2 = \frac{1}{2} I_b \frac{v^2}{r^2} = \frac{1}{2} \frac{I_b}{r^2} (\dot{x}^2 + \dot{y}^2) \quad (6)$$

Dosazením rovnic (5) a (6) do rovnice (4) obdržíme:

$$W_b = \frac{1}{2} m (\dot{x}^2 + \dot{y}^2) + \frac{1}{2} \frac{I_b}{r^2} (\dot{x}^2 + \dot{y}^2) = \frac{1}{2} \left(m + \frac{I_b}{r^2} \right) (\dot{x}^2 + \dot{y}^2) \quad (7)$$

Kinetická energie desky (s momentem setrvačnosti I_p) může být vyjádřena:

$$W_p = \frac{1}{2}(I_p + I_b)(\dot{\alpha}^2 + \dot{\beta}^2) + \frac{1}{2}(\dot{\alpha}x + \dot{\beta}y)^2 \quad (8)$$

Po dosazení rovnic (7) a (8) do rovnice (3) obdržíme výslednou kinetickou energii:

$$W = \frac{1}{2}\left(m + \frac{I_b}{r^2}\right)(\dot{x}^2 + \dot{y}^2) + \frac{1}{2}(I_p + I_b)(\dot{\alpha}^2 + \dot{\beta}^2) + \frac{1}{2}(\dot{\alpha}x + \dot{\beta}y)^2 \quad (9)$$

Potenciální energie kuličky kde h je výška koule vzhledem ke středu desky:

$$V = mgh = mg(x \sin \alpha + y \sin \beta) \quad (10)$$

Zobecněné síly jsou dány točivým momentem generovaným servopohonem včetně přenosového systému:

$$Q_\alpha = F_\alpha d \cos \alpha, \quad Q_\beta = F_\beta d \cos \beta \quad (11)$$

Parametr d je vzdálenost mezi otočným čepem a místem, kde servosystém působí na desku.

Po derivacích potřebných pro získání partikulární formy Euler-Lagrangeových rovnic se získá následující sada nelineárních diferenciálních rovnic

$$x: \left(m + \frac{I_b}{r^2}\right)\ddot{x} - m(\dot{\alpha}\dot{\beta}y + \dot{\alpha}^2x) + mg \sin \alpha = 0 \quad (12)$$

$$y: \left(m + \frac{I_b}{r^2}\right)\ddot{y} - m(\dot{\alpha}\dot{\beta}x + \dot{\beta}^2y) + mg \sin \beta = 0 \quad (13)$$

$$\alpha: (I_p + I_b + mx^2)\ddot{\alpha} + m(\ddot{\beta}xy + \dot{\beta}\dot{x}y + \dot{\beta}x\dot{y} + 2\dot{\alpha}\dot{x}x) + mgx \cos \alpha = F_\alpha d \cos \alpha \quad (14)$$

$$\beta: (I_p + I_b + my^2)\ddot{\beta} + m(\ddot{\alpha}xy + \dot{\alpha}\dot{x}y + \dot{\alpha}x\dot{y} + 2\dot{\beta}\dot{y}y) + mgy \cos \beta = F_\beta d \cos \beta, \quad (15)$$

kde proměnné mají následující význam:

$m(\dot{\alpha}\dot{\beta}y + \dot{\alpha}^2x)$	odstředivý moment vyplývající z otáčení desky,
$(I_b + I_p + mx^2)\ddot{\alpha}$	točivý moment v důsledku setrvačnosti,
$m(\ddot{\beta}xy + \dot{\beta}\dot{x}y + \dot{\beta}x\dot{y})$	gyroskopický vliv,
$2m\dot{\alpha}\dot{x}x$	coriolisovo zrychlení,
$mg \cos \alpha$	gravitační vliv.

Rovnice (12) a (13) popisují pohyb kuličky na desce, říkají, jak zrychlení pohybu kuličky závisí na úhlech a úhlové rychlosti sklonu desky. Rovnice (14) a (15) uvádějí, jak je dynamika sklonu desky ovlivněna vnější hnací silou, polohou a rychlostí kuličky.

2.2.5 Linearizace a zjednodušení modelu

Pro zjednodušení modelu se předpokládá, že krokové motory neztrácejí žádný krok a zatížení neovlivňuje jejich výkon, takže úhly α a β mohou být přímými vstupy systému. To je důvod, proč mohou být rovnice (14) a (15) pro zjednodušení vynechány.

Protože se předpokládá, že kulička je homogenní koule nebo dutá koule se zanedbatelnou tloušťkou pláště, může být její moment setrvačnosti aproximován momentem setrvačnosti koule nebo sférické skořepiny dle rovnic (16).

$$I_{ball} = \frac{2}{5}mr^2 \quad (= I_b) , \quad I_{hollow\ sphere} = \frac{2}{3}mr^2 \quad (16)$$

Pro linearizaci daných rovnic kolem ustáleného stavu, kde jsou úhly α a β nulové, se předpokládá změna v rozsahu $\langle -5^\circ; 5^\circ \rangle$, resp. $|\alpha| \ll 1$ a $|\beta| \ll 1$ v radiánech. Díky tomu mohou být sinusové funkce nahrazeny jejich argumenty. Stejným způsobem se předpokládá, že rychlost změny úhlů se blíží nule. Rovnice (12), (13) obsahují druhou mocninu úhlových rychlostí nebo produkty jejich kombinace, proto tyto prvky mohou být považovány za nulové.

$$\text{Pro } |\alpha| \ll 1, |\beta| \ll 1 \Rightarrow \sin \alpha \cong \alpha ; \sin \beta \cong \beta.$$

$$\text{Pro } |\dot{\alpha}| \ll 1, |\dot{\beta}| \ll 1 \Rightarrow \dot{\alpha}\dot{\beta} \cong 0; \dot{\alpha}^2 \cong 0; \dot{\beta}^2 \cong 0.$$

Rovnice systému po zjednodušení a linearizaci mají následující strukturu:

$$x: \ddot{x} = \frac{5}{7}g \sin \alpha \approx K_b \alpha \quad (17)$$

$$y: \ddot{y} = \frac{5}{7}g \sin \beta \approx K_b \beta \quad (18)$$

Kde K_b je konstanta a závisí pouze na typu koule.

$$K_b = \frac{m}{(m + \frac{I_b}{r^2})} = \frac{m}{(m + \frac{\frac{2}{5}mr^2}{r^2})} = \frac{5}{7} \quad (19)$$

2.2.6 Senzor polohy

Pro určení polohy kuličky se používá systém vidění integrovaný v prostředí Simulinku, který se skládá z kamery Logitech QuickCam Pro 5000 připojené k PC po sběrnici USB. V PC je obrazový signál zpracován příslušným softwarovým vybavením. Vzorkovací interval je nastaven na hodnotu 40 [ms]. Systém vidění je vůči dynamice měřené soustavy dostatečně rychlý, aby diskrétně naměřené hodnoty mohly být považovány za téměř spojité. Dalším parametrem popisujícím měřicí řetězec je převodní konstanta, která signál polohy normalizuje na rozsah [-1; 1] (označené jako MU - Machine Units).

2.2.7 Servosystém modelu

Deska je poháněna dvěma krokovými motory, které mohou být modelovány jako dva intergrátory. Užité krokové motory jsou pro obě osy stejné, jeden krok se rovná konstantní změně úhlu desky. Softwarový ovladač pro řídicí desku krokového motoru a elektroniky desky sám vykazuje další nelineární dynamické chování. Zaprvé, hodnota odeslaná na kartu z prostředí Matlabu je omezena na 1, tj. nelinearita saturace. Pak je hodnota zapsána do vyrovnávací paměti a obsah vyrovnávací paměti nemůže být přepsán, dokud karta nevygeneruje počet pulzů úměrný hodnotě zaznamenané ve vyrovnávací paměti. Tvar vstupního signálu se změní, když se vstup serva změní rychleji než jmenovitá rychlost krokového motoru. Potom motor nebude schopen sledovat vstup a objeví se zpoždění. Vzhledem k výše uvedené nelinearitě není možné linearizovat dynamiku servosystému, tedy nelze použít standardní metody návrhu regulátoru. Řešením je filtrovat vstupní signál tak, že servo může sledovat filtrovaný signál. Nejjednodušší filtr získaný přímo ze známé jmenovité rychlosti krokového motoru je omezovač rychlosti s následující funkcí $u_{\alpha f}(k)$ v diskretním čase s periodou vzorkování T_s [7].

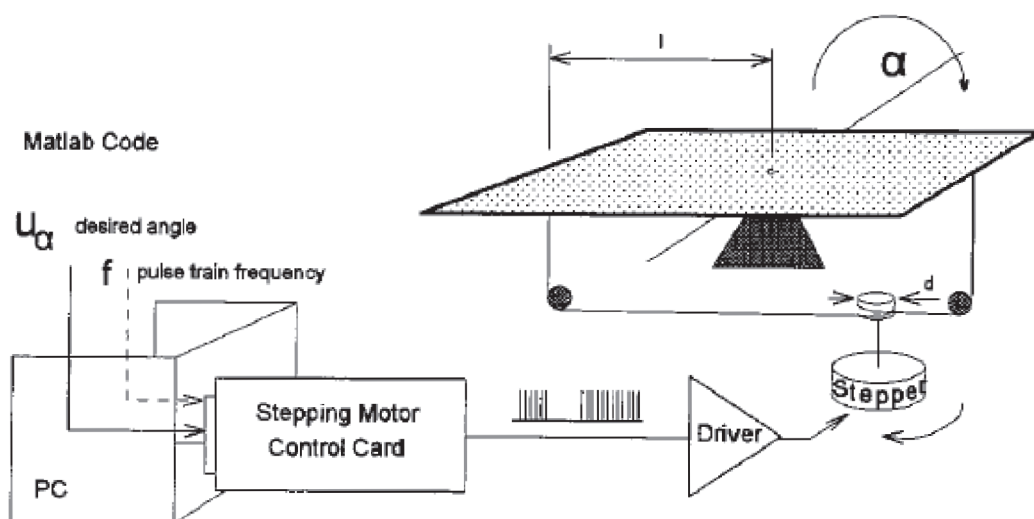
$$rate = \frac{u_{\alpha}(k) - u_{\alpha}(k-1)}{T_s} \tag{20}$$

$$u_{\alpha f}(k) = \begin{cases} +T_s R + u_{\alpha f}(k-1) & \text{pro } rate > +R \\ -T_s R + u_{\alpha f}(k-1) & \text{pro } rate < -R \\ u_{\alpha}(k-1) & \text{jinak} \end{cases}$$

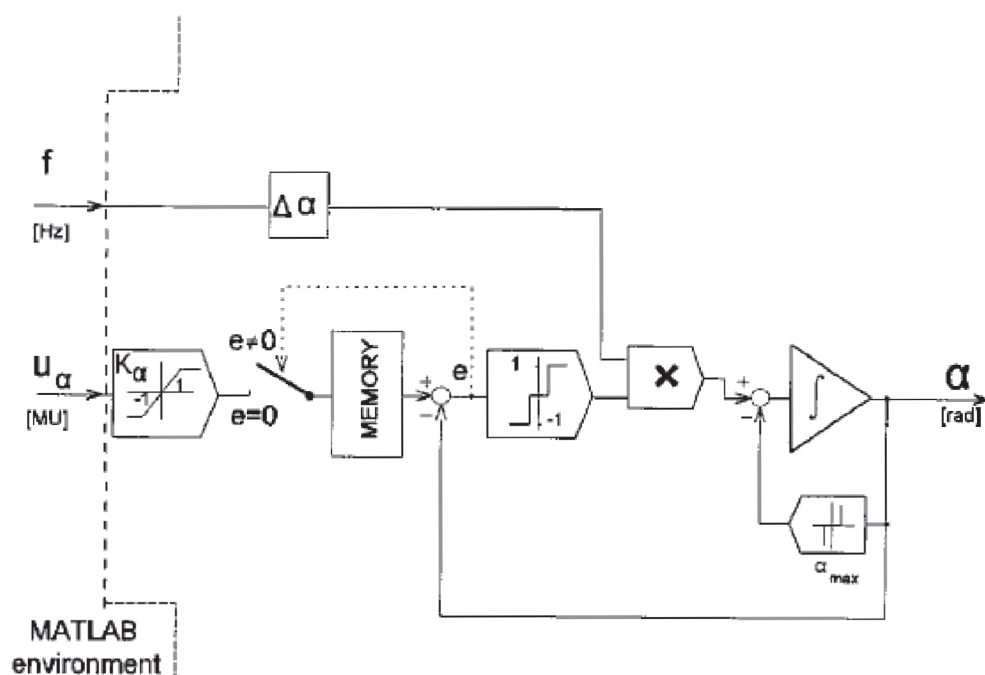
kde u_{α} vstup filtru,
 $u_{\alpha f}$ výstup filtru,
 T_s vzorkovací perioda,
 k index času,
 R prahová hodnota.

Pokud je prahová hodnota pro rychlost R rovna nebo nižší než jmenovitá rychlost servopohonu, servo může kopírovat signál $u_{\alpha f}$ a obsah vyrovnávací paměti řídicího panelu motoru se neustále aktualizuje.

Na obrázcích 4,5 a 6 je uvedené schéma výše popsaného servosystému, jeho blokový diagram a blokové schéma dynamiky nelineárního servosystému.



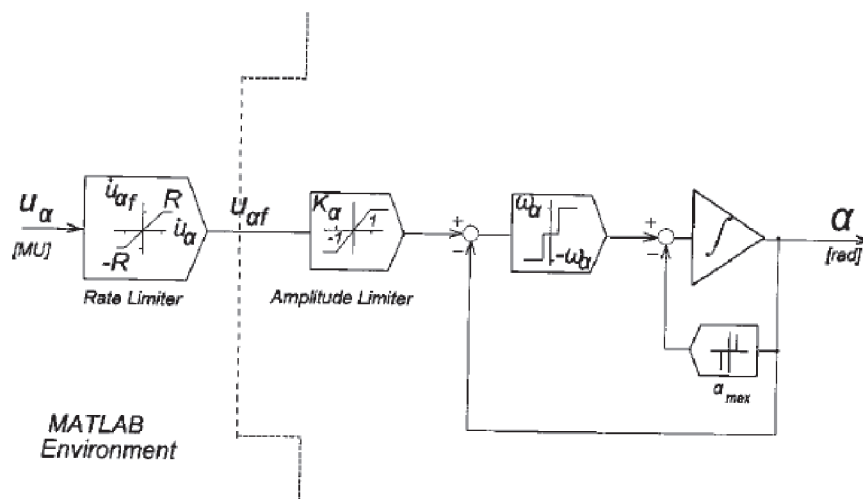
Obr. 4: Schéma servosystému [9]



Obr. 5: Blokový diagram servosystému [9]

Namísto modelování motoru se předpokládá jednoduchá přenosová funkce prvního řádu G_m , což je velmi jednoduchá a spolehlivá aproximace jeho vnitřních funkcí a ovladačů.

$$G_m(s) = \frac{K_m}{T_m s + 1} \quad (21)$$



Obr. 6: Blokové schéma dynamiky nelineárního servosystému [9]

2.3 Stavové řízení

2.3.1 Popis systému ve stavovém prostoru

Stavový prostor je jednou z hlavních metod pro popis chování dynamického systému v teorii řízení. Pohyb systému ve stavovém prostoru ukazuje změnu v jeho stavech.

Ve stavovém prostoru je vytvořen model dynamického systému, který obsahuje sadu vstupních, výstupních a stavových proměnných propojených diferenciálními rovnicemi prvního řádu, které jsou zapsány ve formě matice. Na rozdíl od popisu ve formě přenosové funkce, kde z definice pracujeme s nulovými počátečními podmínkami, umožňuje stavový popis pracovat nejen s lineárními systémy a nulovými počátečními podmínkami. Navíc je ve stavovém prostoru relativně snadné pracovat s MIMO (Multiple-Input Multiple-Output) systémy.

V případě lineárního systému s p vstupy, q výstupy a n stavovými proměnnými je popis:

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \quad (22)$$

$$\mathbf{y}(t) = \mathbf{C}\mathbf{x}(t) + \mathbf{D}\mathbf{u}(t) \quad (23)$$

kde

- $\mathbf{x}(t)$ stavový vektor
- $\mathbf{y}(t)$ výstupní vektor
- $\mathbf{u}(t)$ vstupní vektor
- \mathbf{A} matice systému
- \mathbf{B} matice vstupu
- \mathbf{C} matice výstupu
- \mathbf{D} matice přímé vazby výstupu na vstup

$$\dim[\mathbf{A}] = n \times n, \dim[\mathbf{B}] = n \times p, \dim[\mathbf{C}] = q \times n, \dim[\mathbf{D}] = q \times p, \dot{\mathbf{x}}(t) = \frac{d\mathbf{x}(t)}{dt}$$

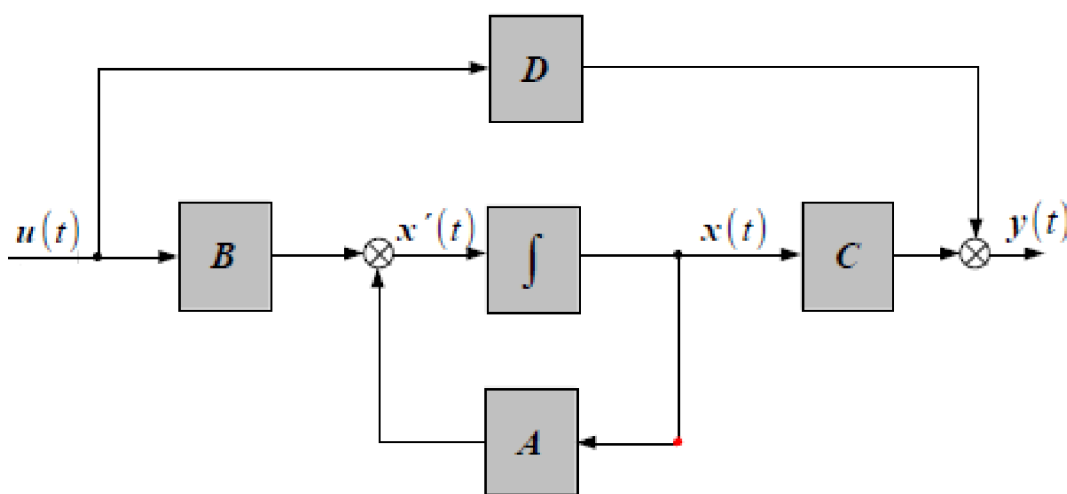
Maticе \mathbf{D} je často nulová, což znamená, že v systému neexistuje žádné explicitní přímé spojení vstupu na výstup.

Pro diskrétní systémy jsou rovnice (22) a (23) ve stavovém prostoru rovnicemi diferenciálními.

$$\mathbf{x}(nT + 1) = \mathbf{A}\mathbf{x}(nT) + \mathbf{B}\mathbf{u}(nT) \quad (24)$$

$$\mathbf{y}(nT) = \mathbf{C}\mathbf{x}(nT) + \mathbf{D}\mathbf{u}(nT) \quad (25)$$

Graficky lze vyjádřit stavovou rovnici dynamiky (22) a výstupní rovnici (23) dle schématu na obrázku 7.



Obr. 7: Bloková schéma spojitého systému ve stavovém prostoru

2.3.2 Stavový popis a přenos systému

Uvažujeme přenos uvedený v manuálu [9], získaný identifikací s úhlem náklonu 0,3 MU:

$$G(s) = \frac{K}{s^2(T_m s + 1)} = \frac{4,803}{s^2(0,187s + 1)} \quad (26)$$

kde

K zesílení

T_m časová konstanta servopohonu

Zavedeme stavové proměnné:

$$\mathbf{x} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \end{bmatrix} = \begin{bmatrix} y_x \\ \dot{y}_x \\ \alpha \end{bmatrix} \quad (27)$$

Stavový popis bude mít tvar:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}_\alpha = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & K \\ 0 & 0 & -\frac{1}{T_m} \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{T_m} \end{bmatrix} \mathbf{u}_\alpha \quad (28)$$

$$\mathbf{y}_x = \mathbf{C}_x = [1 \quad 0 \quad 0] \mathbf{x} \quad (29)$$

kde \mathbf{u}_α požadovaný úhel sklonu
 \mathbf{y}_x poloha kuličky získaná z kamery

Po dosazení známých proměnných z přenosové funkce (26) vyčíslíme (28) následovně.

$$\dot{\mathbf{x}} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 4,803 \\ 0 & 0 & -5,35 \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 \\ 0 \\ 5,35 \end{bmatrix} \mathbf{u}_\alpha \quad (30)$$

$$\mathbf{y}_x = [1 \quad 0 \quad 0] \mathbf{x} \quad (31)$$

2.3.3 Řiditelnost a pozorovatelnost systému

Při analýze systémů ve stavovém prostoru jsou důležité vlastnosti *řiditelnosti* a *pozorovatelnosti* systémů.

Systém popsaný maticemi \mathbf{A} a \mathbf{B} se nazývá plně řiditelný, pokud může být převeden z jakéhokoli počátečního stavu $\mathbf{x}(0)$ do jakéhokoli konečného stavu $\mathbf{x}(t)$ pomocí řídicí veličiny $u(t)$ v konečném čase.

Podmínkou úplné řiditelnosti objektu je rovnost hodnoty jeho matice řiditelnosti \mathbf{R} k řádu systému n . Matice řiditelnosti je vyjádřena prostřednictvím parametrů objektu podle vzorce

$$\mathbf{R} = [\mathbf{B} \mid \mathbf{A}\mathbf{B} \mid \mathbf{A}^2\mathbf{B} \mid \dots \mid \mathbf{A}^{n-1}\mathbf{B}]. \quad (32)$$

Pro řízení systému je nutné mít informace o jeho aktuálním stavu, tj. o hodnotách stavového vektoru $\mathbf{x}(t)$ v každém časovém okamžiku. Některé stavové proměnné jsou však abstraktní, nemají žádné fyzikální analogie v reálném systému, nebo je nelze měřit. Fyzické výstupní proměnné $y(t)$ jsou měřitelné a pozorovatelné.

Stav $\mathbf{x}(t)$ se nazývá pozorovatelný, jestliže v době pozorování $t = t_0$ můžeme jednoznačně určit $\mathbf{x}(t_0)$ měřením vstupu $u(t)$ a výstupu $y(t)$ proměnných v časovém intervalu.

Podmínkou pozorovatelnosti stavu spojitého a časově invariantního systému (LTI systémy) je rovnost hodnoty jeho matice pozorovatelnosti \mathbf{P} k řádu systému n . Matice pozorovatelnosti je vyjádřena prostřednictvím parametrů objektu podle vzorce

$$\mathbf{P}^T = [\mathbf{C}^T \mid \mathbf{A}^T\mathbf{C}^T \mid (\mathbf{A}^T)^2\mathbf{C}^T \mid \dots \mid (\mathbf{A}^T)^{n-1}\mathbf{C}^T]. \quad (33)$$

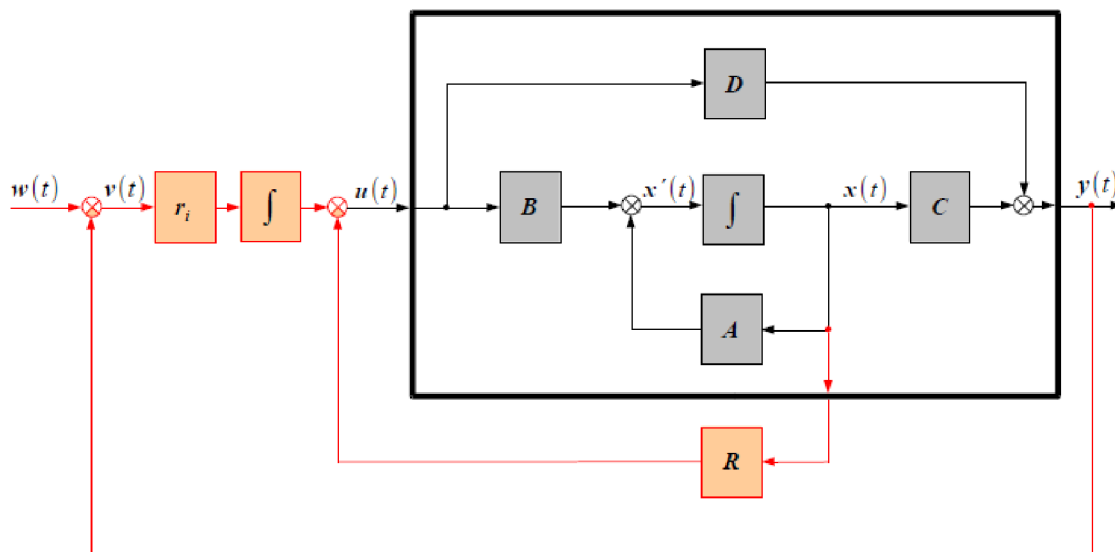
Vlastnosti řídicího a pozorovatelného systému musí být zvažovány společně, aby byl problém s regulací správně položen a zásadně řešitelný.

Pro automatizaci studia řídicích systémů na říditelnost a pozorovatelnost v programovacím systému Matlab existují funkce *ctrb* a *obsv*.

2.3.4 Stavový zpětnovazební regulátor s integrátorem

Cílem návrhu je použitím metody umístění pólů určit zesílení \mathbf{R} , \mathbf{r}_i pro žádaná vlastní čísla c stavového modelu uzavřené smyčky.

Grafické znázornění stavového zpětnovazebního regulátoru s integrátorem je na obrázku 8.



Obr. 8: Schéma stavového zpětnovazebního regulátoru s integrátorem [11]

Použitý zákon řízení:

$$\mathbf{u} = -\mathbf{R}\mathbf{x} + \mathbf{r}_i \int_0^t (\mathbf{w} - \mathbf{y})d\tau \quad (34)$$

Přepíšeme do tvaru:

$$\mathbf{u} = -\mathbf{R}\mathbf{x} + \mathbf{r}_i\mathbf{v} \quad (35)$$

kde $\mathbf{v} = \int_0^t (\mathbf{w} - \mathbf{y})d\tau$, tj. $\dot{\mathbf{v}} = \mathbf{w} - \mathbf{y}$, což představuje nové stavové rovnice.

Zákon řízení teď můžeme zapsat jako:

$$\mathbf{u} = -\mathbf{R}\mathbf{x} + \mathbf{r}_i\mathbf{v}$$

$$\dot{\mathbf{v}} = \mathbf{w} - \mathbf{y}.$$

Ted je třeba zpracovat stavový model uzavřené smyčky s novým stavovým vektorem $[\mathbf{x} \ \mathbf{v}]^T$:

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} \quad (36)$$

$$\dot{\mathbf{v}} = \mathbf{w} - \mathbf{y} = \mathbf{w} - (\mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u}) \quad (37)$$

Přítom

$$\mathbf{u} = -\mathbf{R}\mathbf{x} + \mathbf{r}_1\mathbf{v} = \begin{bmatrix} -\mathbf{R} & \mathbf{r}_1 \end{bmatrix} \begin{bmatrix} \mathbf{x} \\ \mathbf{v} \end{bmatrix} \quad (38)$$

Stavová matice uzavřené smyčky tak koresponduje s maticí $\mathbf{X-YZ}$ pro

$$\mathbf{X} = \begin{bmatrix} \mathbf{A} & 0 \\ -\mathbf{C} & 0 \end{bmatrix}, \quad \mathbf{Y} = \begin{bmatrix} \mathbf{B} \\ -\mathbf{D} \end{bmatrix}, \quad \mathbf{Z} = \begin{bmatrix} \mathbf{R} & -\mathbf{r}_1 \end{bmatrix} \quad (39)$$

2.4 Metoda geometrického místa kořenů

Relativní stabilita a kvalita přechodného režimu uzavřeného řídicího systému přímo souvisí s polohou kořenů jeho charakteristické rovnice na s -rovině. Pro zajištění správného umístění těchto kořenů je často nutné nakonfigurovat jeden nebo více systémových parametrů. Proto má smysl zkoumat, jak se kořeny charakteristické rovnice pohybují v s -rovině jako parametry změny systému; jinými slovy, zajímavé jsou trajektorie kořenů na s -rovině. Kořenový hodograf byl navržen Evansem v roce 1948 a následně rozšířen ve strojírenské praxi. Umožňuje inženýrům posoudit citlivost pólů systému na změny v jakémkoliv parametru. Metoda kořenového hodografu přináší největší výhodu v kombinaci s Routh-Hurwitzovým kritériem.

Metoda kořenového hodografu je grafická a samotný hodograf poskytuje kvalitativní informace o stabilitě a dynamickém výkonu systému. To platí se stejným úspěchem jak pro systémy s jednou smyčkou, tak pro více smyčky. Pokud poloha kořenů charakteristické rovnice z nějakého důvodu nevyhovuje konstruktérovi, může snadno určit z kořenového hodografu, jak změnit proměnný parametr systému.

2.4.1 Teoretické poznámky

Kořenový hodograf je soubor trajektorií pohybu všech kořenů charakteristické rovnice uzavřeného systému při změně jakéhokoliv parametru tohoto systému.

Přenosová funkce uzavřeného systému

$$W(s) = K \frac{A(s)}{B(s)} = K \frac{(s - s_1)(s - s_2) \dots (s - s_m)}{(s - p_1)(s - p_2) \dots (s - p_n)} \quad (40)$$

kde s_i nuly přenosové funkce,
 p_i póly přenosové funkce.

Přičemž řád polynomu čitatele je m , řád polynomu jmenovatele je n , $m \leq n$ pro fyzicky realizovatelné systémy.

Metoda kořenového hodografu spojuje dynamické charakteristiky systému s chováním nul a pólů jeho přenosové funkce, které získáme z nul a pólů systému s otevřenou smyčkou při změně nějakého parametru (obvykle zesílení systému

s otevřenou smyčkou). Uzavřený systém je propojen s otevřením pomocí následujícího vztahu:

$$W(s) = \frac{W_p}{1 + W_o}, \quad (41)$$

kde W_p funkce přímého přenosu systému
 W_o funkce přenosu s otevřenou smyčkou

Tento vzorec platí pouze pro zápornou zpětnou vazbu, jinak bude znaménko za jednotkou záporné.

Nechť bod p je pól uzavřeného systému. Nakreslíme do tohoto bodu vektory ze všech nul systému otevřené smyčky W_p (označíme argumenty těchto vektorů θ_j^0) a všechny póly W_p (označujeme argumenty těchto vektorů θ_j^p). Pak bude kořenovým hodografem místo bodů, které splňují následující rovnici:

$$\sum_{j=1}^n \theta_j^0 - \sum_{j=1}^n \theta_j^p = \pm(2u + 1)\pi, \quad u = 0, 1, 2 \dots \quad (42)$$

Průsečík asymptot je definován jako těžiště souřadnic nul a pólů

$$\sigma_\alpha = \frac{\sum_{i=1}^n p_i - \sum_{j=1}^n s_j}{n - m} \quad (43)$$

Úhel sklonu asymptot je určen vzorcem

$$Q = \frac{\pm\pi(2u - 1)}{n - m} \quad (44)$$

Metoda kořenového hodografu umožňuje zvolit zesílení systému řízení, zvolit umístění nul a pólů korekčních členů řídicích systému.

2.4.2 Vlastnosti metody geometrického místa kořenů

Podíváme se na vlastnosti kořenového hodografu při změně zesílení:

1. Větve kořenového hodografu jsou spojitě a symetrické kolem reálné osy komplexní roviny.
2. Počet větví kořenového hodografu se rovná řadu systému n .
3. Větve začínají na pólech systému s otevřenou smyčkou (protože s nulovým zesílením K se póly systémů s otevřenou smyčkou a uzavřenou smyčkou shodují). S rostoucím K od nuly do nekonečna se póly uzavřeného systému pohybují podél větví kořenového hodografu.
4. Jelikož při K se póly uzavřeného systému stávají rovnými nulám otevřeného systému, m větví kořenového hodografu končí na nulách uzavřeného systému a ostatní větve směřují do nekonečna.

5. Uzavřený systém je stabilní, pokud jeho póly leží v levé polovině. Při přechodu imaginární osy větvemi hodografu zleva doprava se tedy systém stává ze stabilního nestabilním. Zesílení odpovídající tomuto přechodu se nazývá kritickým. Tato vlastnost je užitečná při posuzování stability systému.

3 PRAKTICKÁ ČÁST

3.1 Seznámení s modelem

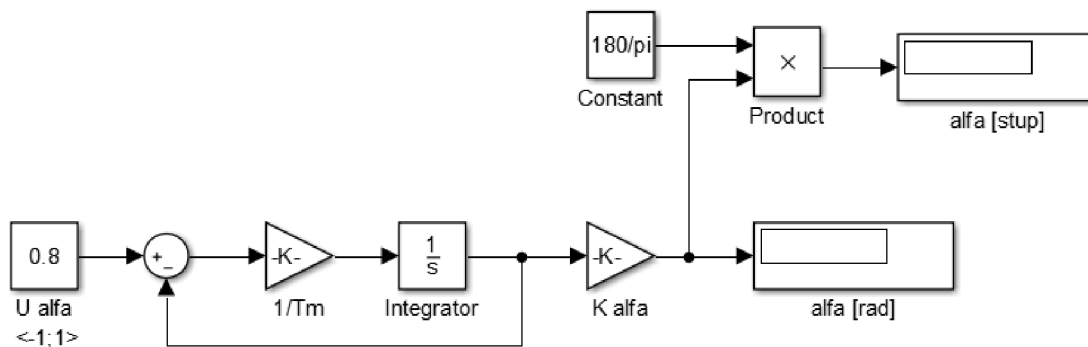
Na obrázku vidíme, jak vypadá skutečný model zařízení a jak vypadá obraz snímáný kamerou. Při zahájení práce se systémem byla kamera umístěna ve vhodné výšce, aby se mohla natáčet celá deska a zároveň byl odstraněn vliv okolního prostředí na minimum. Nutností bylo dosáhnout viditelnosti kuličky ve všech bodech na ploše. V opačném případě by byla kulička pouze částečně viditelná a kamera by nebyla schopna rozpoznat její polohu.



Obr. 9: Pohled na model[8] a obraz z kamery

Pro první seznámení s modelem byl proveden jednoduchý experiment pro porovnání parametrů reálného modelu s parametry uvedenými v manuálu CE151.

O naklání plochy se stará dvojice krokových motorů, viz kap. 2, takže existuje závislost úhlu naklonění roviny na vstupním napětí krokového motoru. V manuálu [9] se tato závislost vyjadřuje jako statické zesílení servosystému K_a . Za účelem nalezení této závislosti byla vytvořeno schéma v Simulinku, která je zobrazeno na obrázku 10. Schéma převádí vstupní napětí motoru z jednotek $[-1;1]$ MU na úhel sklonu desky ve stupních a radiánech. Získané hodnoty jsou uvedeny v tabulce 1.



Obr. 10: Schéma převodu napětí na úhel

Tab. 1: Simulované hodnoty

MU	α [rad]	α [°]
-0,8	-0,1495	-8,567
-0,7	-0,1308	-7,496
-0,6	-0,1121	-6,452
-0,5	-0,09345	-5,354
-0,4	-0,07476	-4,284
-0,3	-0,05607	-3,213
-0,2	-0,03738	-2,142
-0,1	-0,01869	-1,071
0	0	0
0,1	0,01869	1,071
0,2	0,03738	2,142
0,3	0,05607	3,213
0,4	0,07476	4,284
0,5	0,09345	5,354
0,6	0,1121	6,452
0,7	0,1308	7,496
0,8	0,1495	8,567

Také pro kontrolu simulovaných dat byl úhel sklonu desky měřen manuálně pomocí úhlooměru. Získané hodnoty jsou uvedeny v tabulce 2. Pro přehlednost ze získaných dat byl vytvořen graf, viz obrázek 11.

Jak je vidět z grafu 11, závislost je lineární a rozdíl mezi simulací a reálným měřením je minimální. Osy x a y jsou symetrické, takže není třeba provádět podobná měření pro osu y . Dál je možno snadno vypočítat koeficient K_α podle vzorce:

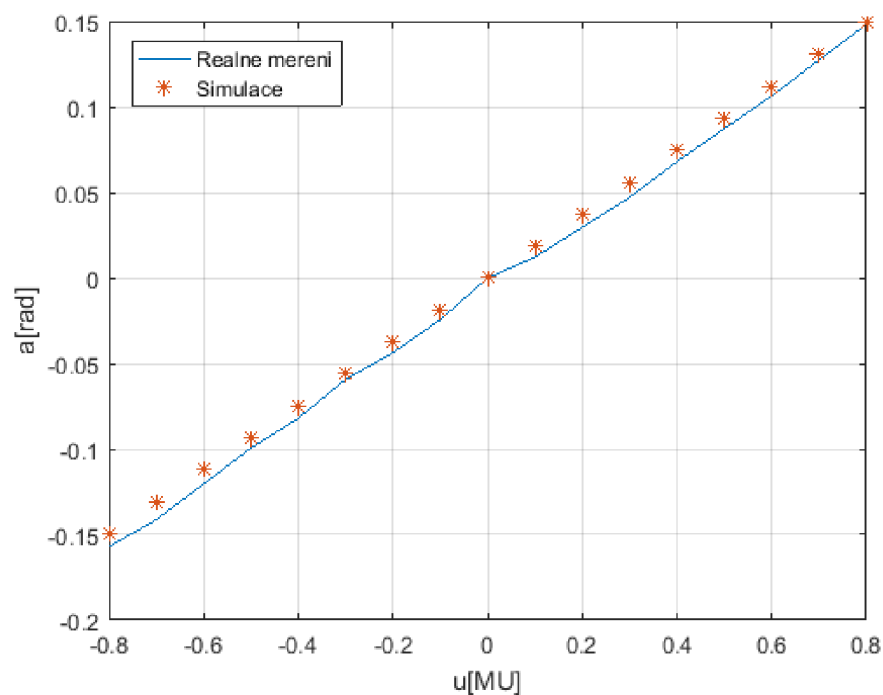
$$\alpha = K_\alpha * u \quad (45)$$

$$K_\alpha = \frac{\alpha}{u} = 0,2 \quad (46)$$

Přičemž v manuálu $K_\alpha = 0,187$ [rad/MU].

Tab. 2: Naměřené hodnoty

MU	α [rad]	α [°]
-0,8	-0,15708	-9
-0,7	-0,14137	-8,1
-0,6	-0,12043	-6,9
-0,5	-0,09948	-5,7
-0,4	-0,08203	-4,7
-0,3	-0,05934	-3,4
-0,2	-0,04363	-2,5
-0,1	-0,02443	-1,4
0	0	0
0,1	0,012217	0,7
0,2	0,029671	1,7
0,3	0,047124	2,7
0,4	0,068068	3,9
0,5	0,087266	5,0
0,6	0,106465	6,1
0,7	0,127409	7,3
0,8	0,148353	8,5

Obr. 11: Závislost náklonu α [rad] na změně vstupního napětí u [MU] pro osu x

3.2 Identifikace

Pro další práci se systémem byla nutná identifikace tohoto systému. Identifikace byla provedena podle následujícího postupu:

- umístěte desku do vodorovné polohy,
- kuličku umístěte v rohu desky,
- spusťte proceduru Matlab pro uložení historie pozice míče,
- nakloňte desku na určitý úhel a zaznamenejte pohyb koule.

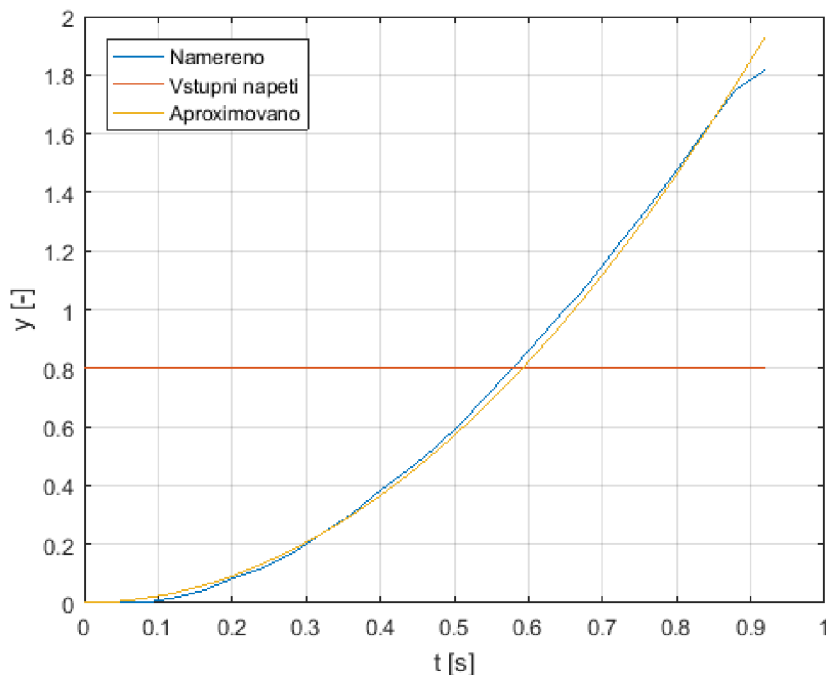
Postupně byly získány přechodné odezvy pro obě osy. Deska byla nakloněna v úhlech 0,8; 0,6; 0,4 a 0,2, což odpovídá 80%, 60%, 40% a 20% maximálního náklonu. Kulička se pouštěla po desce s ustáleným úhlem, její pohyb byl fixován kamerou a uložen. Po prvním dotyku hrany desky se kulička odrazila. Pro identifikaci byla zvolena data od začátku pohybu kuličky, když ještě měla nulovou počáteční rychlost, až do prvního kontaktu s okrajem desky. Pro každý náklon bylo provedeno několik měření, z nichž byly vybrány nejúspěšnější pokusy. Pro provedení identifikace bylo upraveno dodávané schéma v Simulinku.

Během experimentu byl objeven problém se získáváním informací z kamery. Pro tento účel byl použit blok “Analog Input” v Simulinku poskytnutý výrobcem. Tento blok čte obraz z kamery a pomocí algoritmu najde souřadnice koule, které jsou dále zobrazeny na displeji a slouží k vytvoření cesty míče. Tento blok je napsán v jazyce C a je uživatelsky uzavřen k editaci, tj. neumožňuje měnit vnitřní strukturu a parametry. Dají se měnit jen parametry nastavení kamery. Perioda vzorkování byla nastavena na 0,04 [s], snímková frekvence na 25[fps]. Poznamenejme, že při rychlém pohybu kuličky během snímání docházelo ke ztrátě snímaného vzorku. Proto bylo dost obtížné dostat vhodná data pro další identifikaci systému.

V dalších tezích na podobné téma [2,5] bylo nalezeno řešení použít vršek od hnědé elektrikařské krycí lišty a tím vytvořit míčku kolejnici, aby nepřeskakoval z jedné osy do druhé, a nasnímaná dráha byla přesnější. Vlastní experimenty s modelem však nevyžadovaly použití tohoto řešení. Pokud je kulička umístěna do rohu desky, pak její pohyb na jedné straně bude omezen okrajem desky. Kulička se bude posouvat podél desky v přímé linii, aniž by ovlivnila druhou osu.

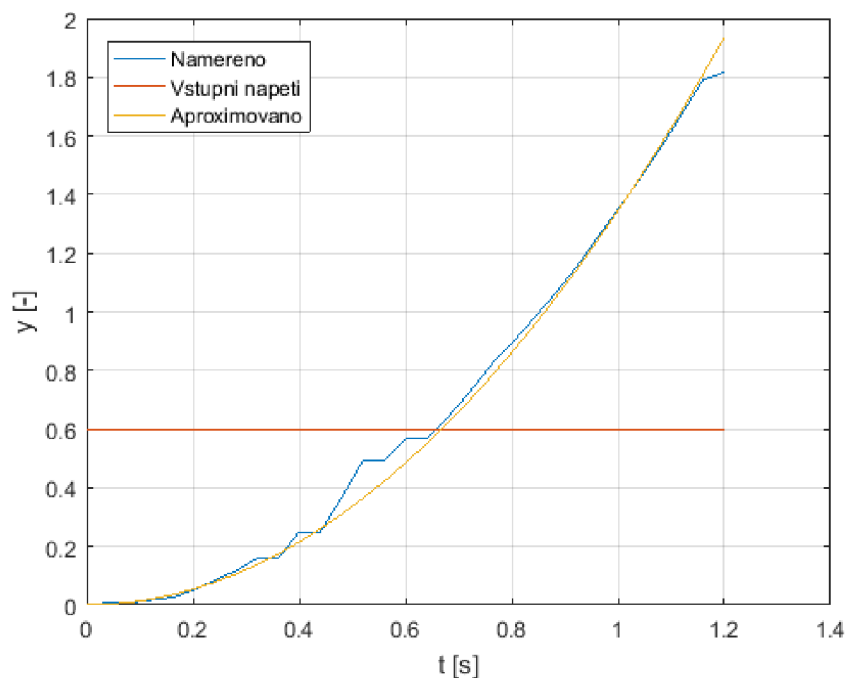
Pro účely identifikační byl vytvořen program v Matlabu. Program načítá naměřená data. Výsledkem programu je aproximační graf vstupních dat a přenosová funkce. Identifikace byla provedena pomocí funkce “*fminsearch*”. Tato funkce byla zvolena protože je velmi jednoduchá a flexibilní.

Následující obrázky ukazují zaznamenané dráhy kuličky pro osy x a y pro různé úhly náklonu posunuté do nuly a jejich aproximaci. Pod každým obrázkem je uveden příslušný identifikovaný přenos získaný aproximací naměřených dat.



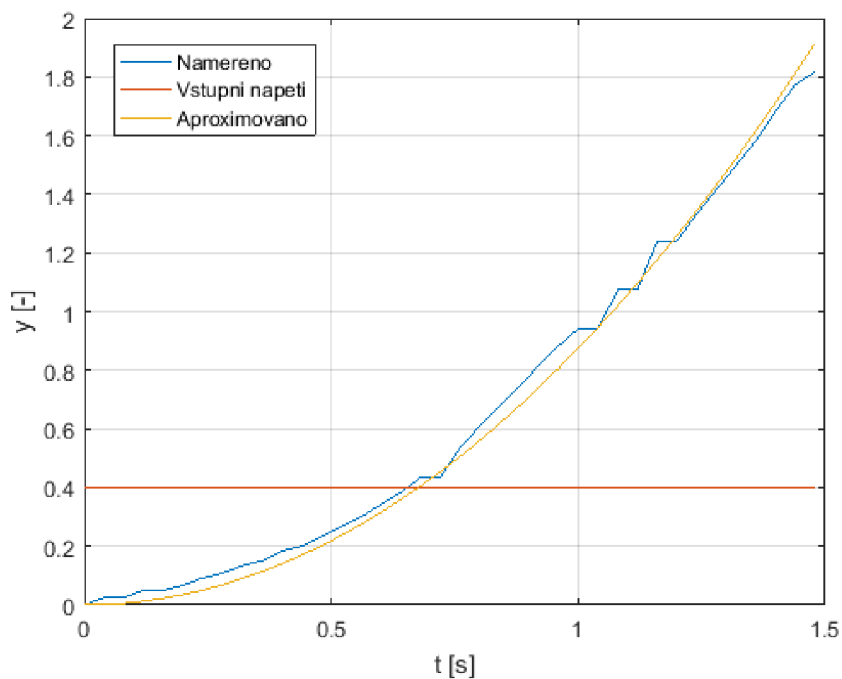
Obr. 12: Aproximovaný průběh polohy míčku pro náklon 0,8 MU v ose x

$$G_{x0.8}(s) = \frac{5,706}{4,074 * 10^{-9}s^3 + s^2} \quad (47)$$



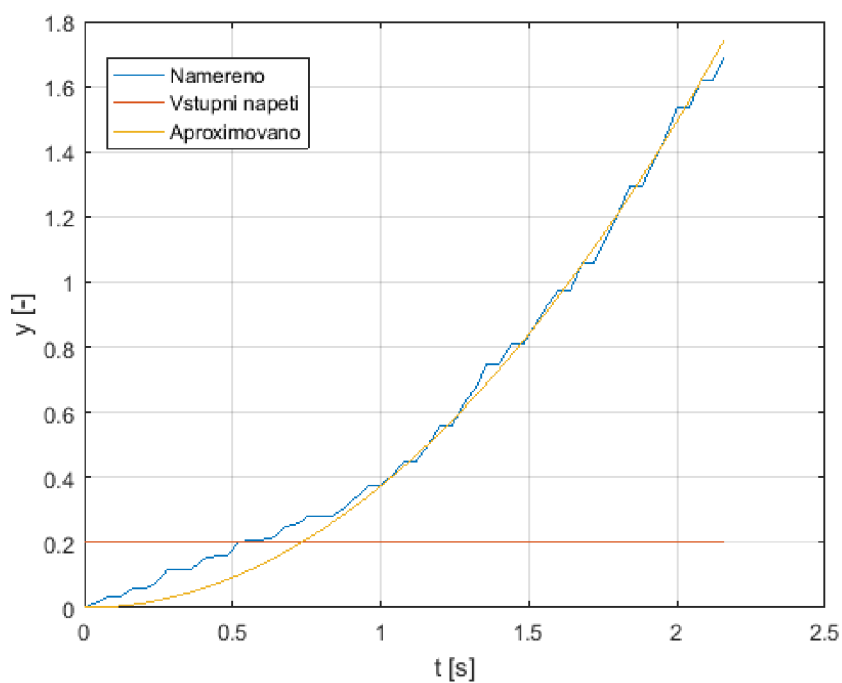
Obr. 13: Aproximovaný průběh polohy míčku pro náklon 0,6 MU v ose x

$$G_{x0.6}(s) = \frac{4,484}{9 * 10^{-9}s^3 + s^2} \quad (48)$$



Obr. 14: Aproximovaný průběh polohy míčku pro náklon 0,4 MU v ose x

$$G_{x0.4}(s) = \frac{4,3732}{1,082 * 10^{-8}s^3 + s^2} \quad (49)$$

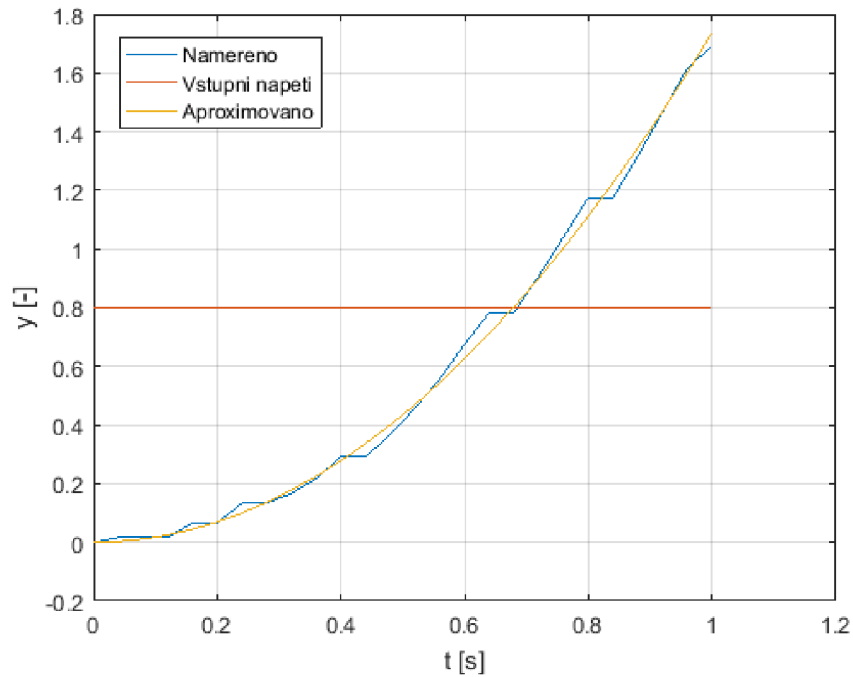


Obr. 15: Aproximovaný průběh polohy míčku pro náklon 0,2 MU v ose x

$$G_{x0.2}(s) = \frac{3,735}{5,351 * 10^{-8}s^3 + s^2} \quad (50)$$

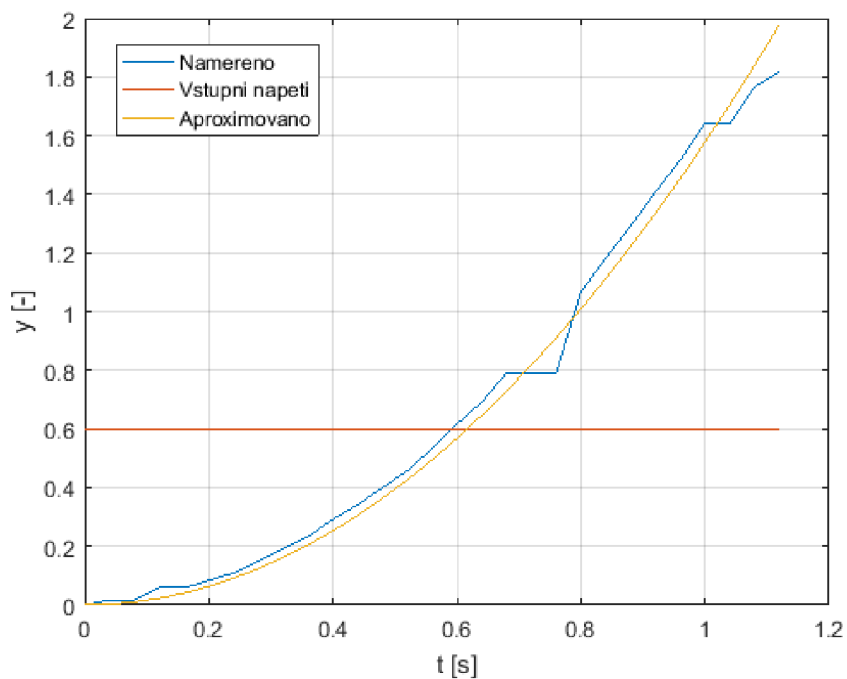
Výsledný přenos $G_x(s)$ pro osu x byl získán vypočtením aritmetického průměru přenosů (47), (48), (49), (50) získaných pro uvedené čtyři úhly náklonu.

$$G_x(s) = \frac{4,5746}{1,9351 * 10^{-8}s^3 + s^2} \quad (51)$$



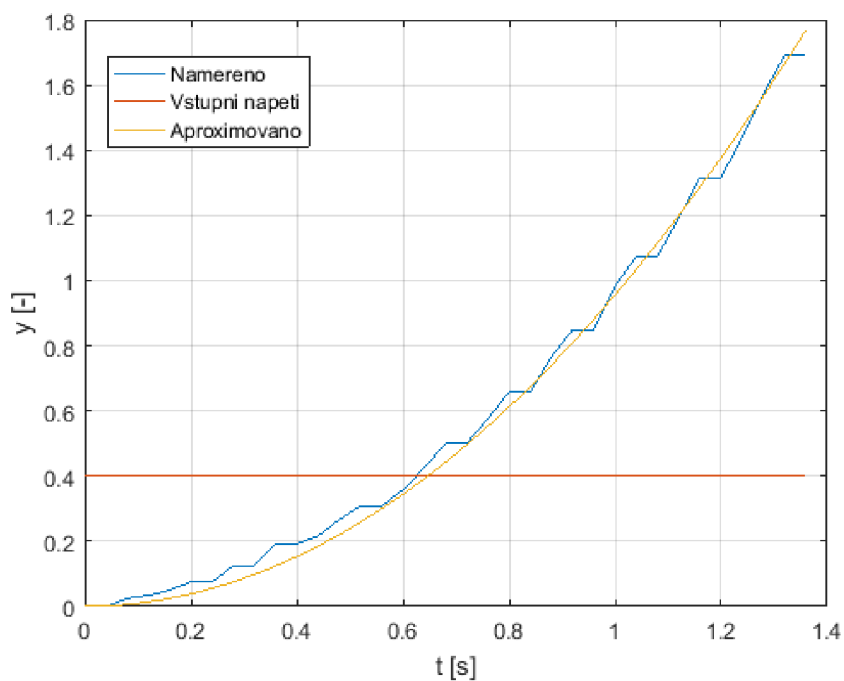
Obr. 16: Aproximovaný průběh pro náklon 0,8 MU v ose y

$$G_{y0.8}(s) = \frac{4,343}{7,214 * 10^{-8}s^3 + s^2} \quad (52)$$



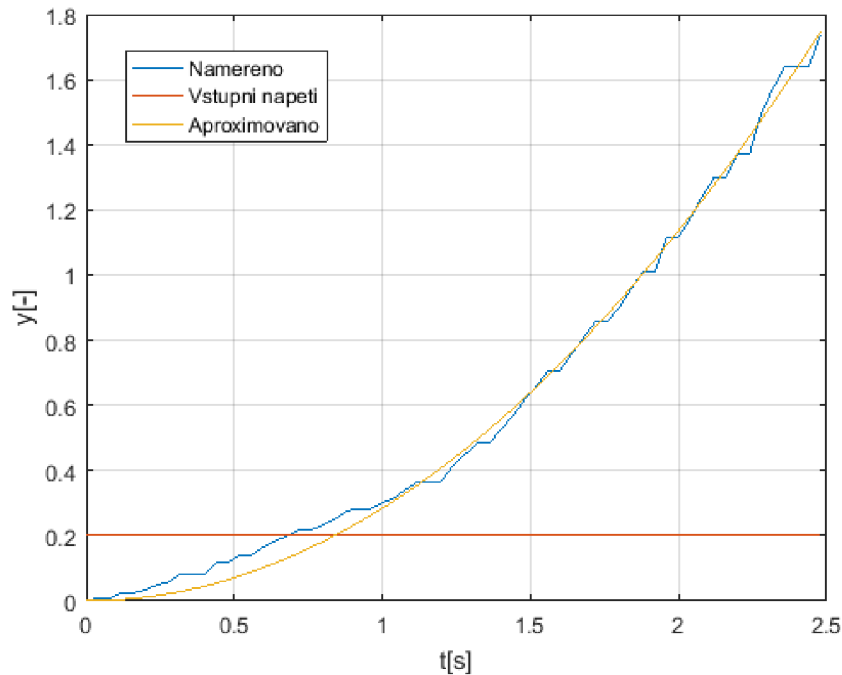
Obr. 17: Aproximovaný průběh pro náклон 0,6 MU v ose y

$$G_{y0.6}(s) = \frac{5,259}{2,541 * 10^{-8}s^3 + s^2} \quad (53)$$



Obr. 18: Aproximovaný průběh pro náклон 0,4 MU v ose y

$$G_{y0.4}(s) = \frac{4,781}{1,555 * 10^{-8}s^3 + s^2} \quad (54)$$



Obr. 19: Aproximovaný průběh pro náklon 0,2 MU v ose y

$$G_{y0.2}(s) = \frac{2,842}{3,006 * 10^{-7}s^3 + s^2} \quad (55)$$

Výsledný přenos $G_y(s)$ pro osu y byl získán vypočtením aritmetického průměru přenosů (52), (53), (54), (55) získaných pro uvedené čtyři úhly náklonu.

$$G_y(s) = \frac{4,3063}{1,0343 * 10^{-7}s^3 + s^2} \quad (56)$$

Přičemž manuál CE151 obsahuje následující přenosovou funkci:

$$G(s) = \frac{4,803}{0,187s^3 + s^2} \quad (57)$$

Nalezený koeficient zesílení systému K odpovídá údajům v manuálu, ale časová konstanta T_m ve jmenovateli v přenosových funkcích obou os má tendenci k 0.

3.3 Návrh regulátoru

3.3.1 Metoda geometrického místa kořenu

Pro nalezení nul a pólů systému v Matlabu existuje funkce „*rootlocus*“. Pro nastavení koeficientů PID regulátoru byl použit Control System Toolbox v MATLABu. Jedná se o kolekci matematických algoritmů pro modelování, analýzu a návrh systémů automatického řízení. Interaktivní prostředí tohoto toolboxu poskytuje uživateli grafické ovládací prvky pro současné zobrazení různých odezev systému a přepínání mezi nimi.

To umožňuje vypočítat různé charakteristiky odezvy, jako je změna přechodové doby, přeregulování, doba náběhu, rezerva stability v amplitudě a fázi.

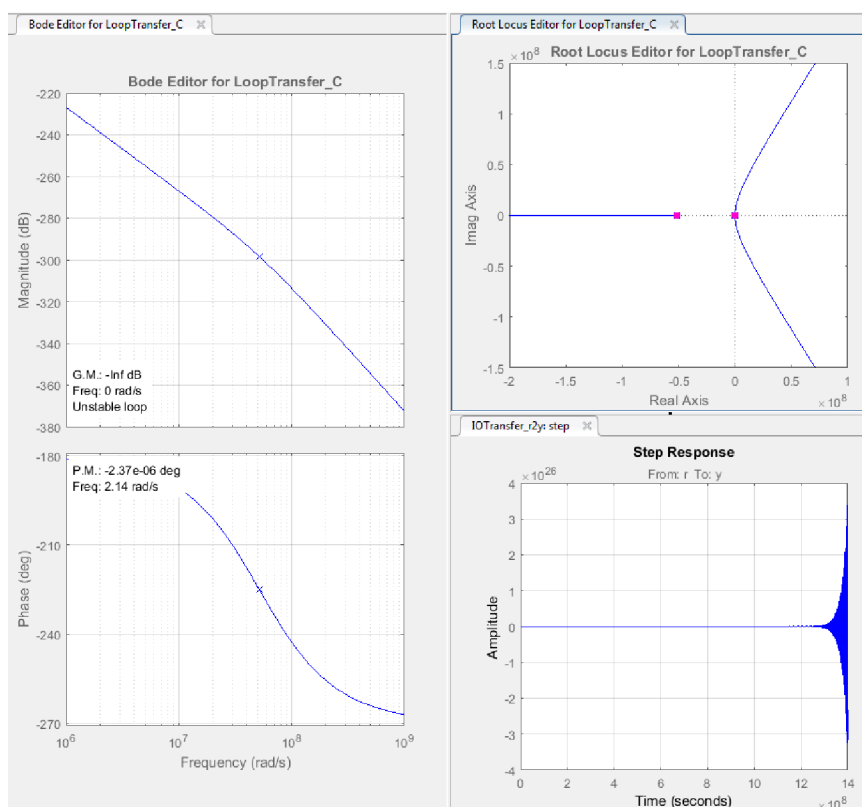
Pro přenos systému v ose x (51) byly v Matlabu pomocí Control System Toolbox nakresleny kořenový hodograf, přechodová charakteristika a diagram *bode*.

Z grafů 20 je zřejmé, že původní systém je nestabilní. Dvě větve kořenového hodografu mají začátek na nule a směřují do nestabilní oblasti, proto není vhodné použít P nebo PI regulátor pro řízení – kvůli přidání dalšího pólu větve kořenového hodografu se nedostanou do stabilní oblasti. Stabilitu systému lze dosáhnout přidáním nuly, což vede k rozhodnutí použít PD nebo PID regulátor.

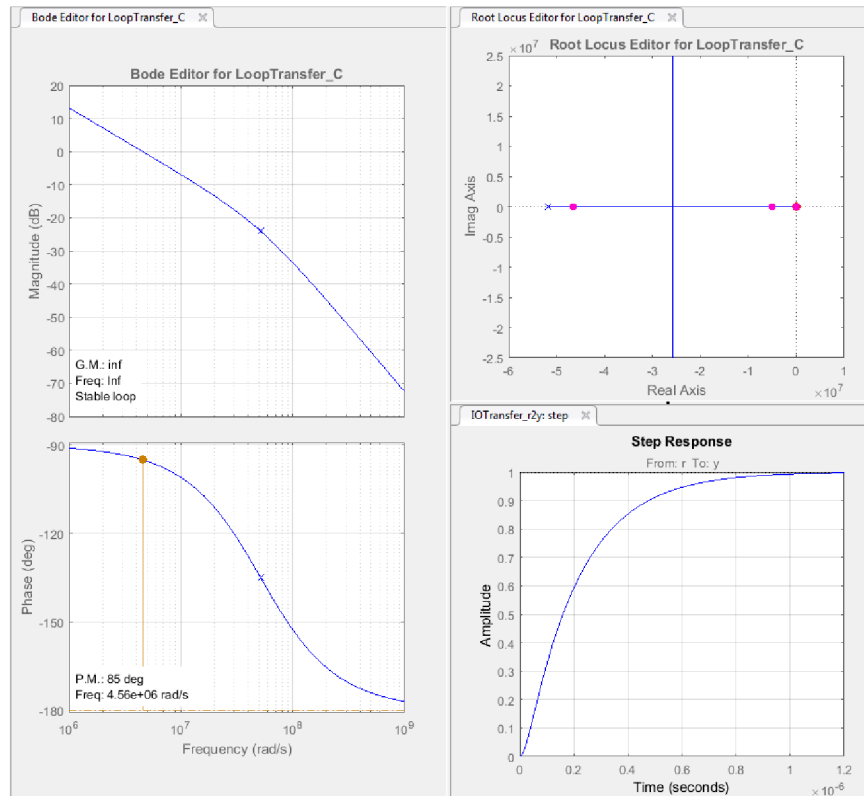
PD regulátor představuje v otevřené smyčce nulu a zesílení K . Na obrázku 21 vidíme grafy systému po přidání nuly. Hodnota nuly byla vybrána jako $-1,9351 \cdot 10^{-8}$ pro kompenzaci časové konstanty ve jmenovateli. Tato nula posunula větve kořenového hodografu do stabilní oblasti. Doba přechodu je 1,2 [s], bez překmitu.

Přenosová funkce pro navržený PD regulátor je uvedena vztahem (58).

$$G_{PDx}(s) = 9,9981 \cdot 10^5 (s + 1,935 \cdot 10^{-8}) \quad (58)$$



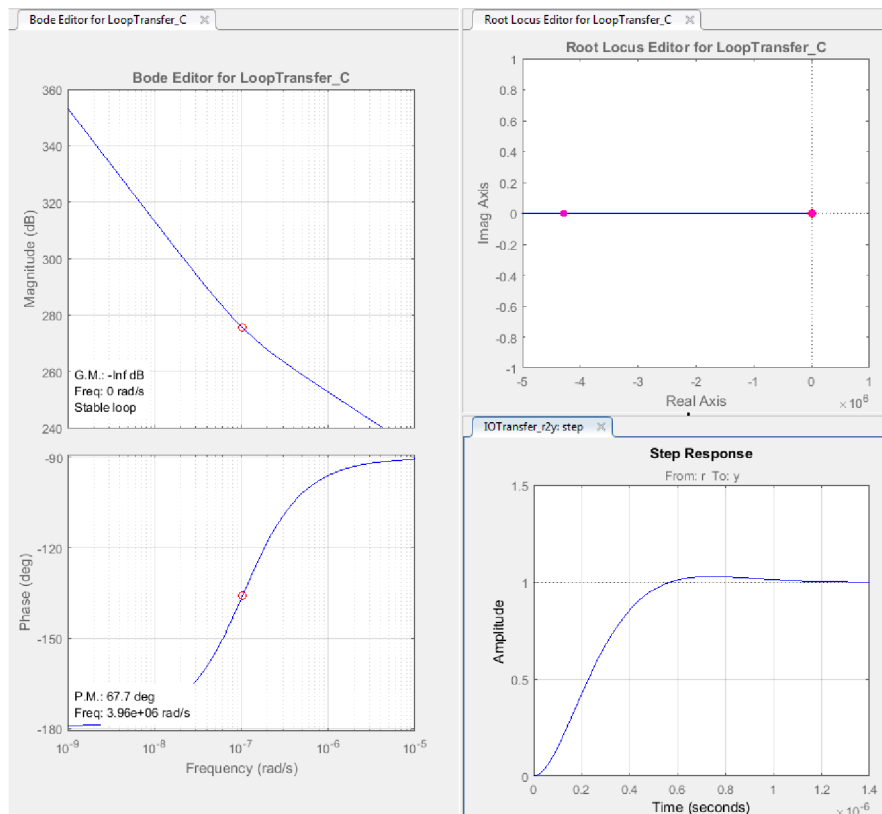
Obr. 20: Grafy neregulovaného systému pro osu x

Obr. 21: Grafy pro osu x s PD regulátorem

Ideální PID regulátor reprezentuje v otevřené smyčce pól v hodnotě nula, dvě nuly a zesílení K . Pro řízení bylo rozhodnuto zvolit PD regulátor, protože přidání pólu k získání PID regulátoru výrazně zhoršilo charakteristiku systému.

Stejný postup byl použit pro osu y . Přenosová funkce PD regulátoru pro osu y je uvedena vztahem (59).

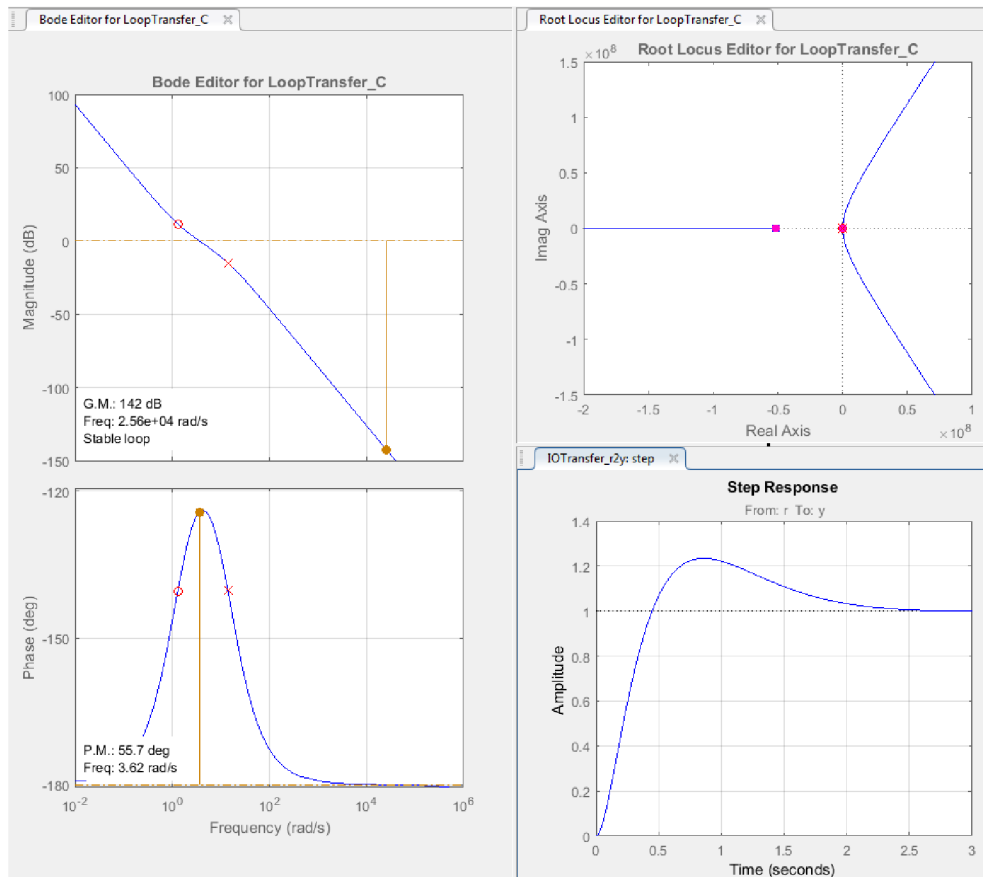
$$G_{PDy}(s) = 9,9469 * 10^5 (s + 1,034 * 10^{-7}) \quad (59)$$



Obr. 22: Grafy pro osu y s PD regulátorem

Na obrázcích 21 a 22 – na přechodových charakteristikách, je vidět dobu přechodu 1,2 a $1,4 \cdot 10^{-6}$ [s], což reálný systém nedokáže zpracovat. Proto byl vypočítán regulátor s ohledem na vlastnosti skutečného modelu. Řešením bylo použít filtraci regulátoru, které se docílilo tak, že kromě nuly se přidal pól, odpovídající dané filtraci. Za předpokladu, že osy jsou symetrické, je regulátor představen pouze pro osu x. Nula byla umístěna v bodě -1.3, potom pól daný filtrací by měl být 10 až 15 krát větší, tedy -14. Na obrázku 23 jsou zobrazeny odezvy systému v ose x s PD regulátorem s filtrací. Doba ustálení je 2,5 [s], překmit 12,2 [%]. Přenosová funkce PD regulátoru s filtrem je:

$$G_{PD}(s) = \frac{10,769(s + 1,3)}{(s + 14)} \quad (60)$$

Obr. 23: Grafy pro osu x s PD regulátorem s filtrací

3.3.2 Metoda stavového prostoru

K regulaci systému pomocí metody stavového prostoru byl napsán kód v Matlabu. Kód obsahuje odpovídající matice systému, kontrolu říditelnosti a pozorovatelnosti, převod systému na diskrétní a další výpočty popsané v teoretické části této práce. Výsledkem programu jsou hodnoty regulátoru \mathbf{R}_i a integrátoru \mathbf{r}_i a graf přechodové charakteristiky zregulovaného systému.

Protože osy jsou symetrické, pro tuto metodu budou zobrazeny výsledky pouze pro jednu osu x . Přenos soustavy pro osu x je uvažován ve tvaru (51). Model systému ve stavovém prostoru podle (28), (29) bude následující:

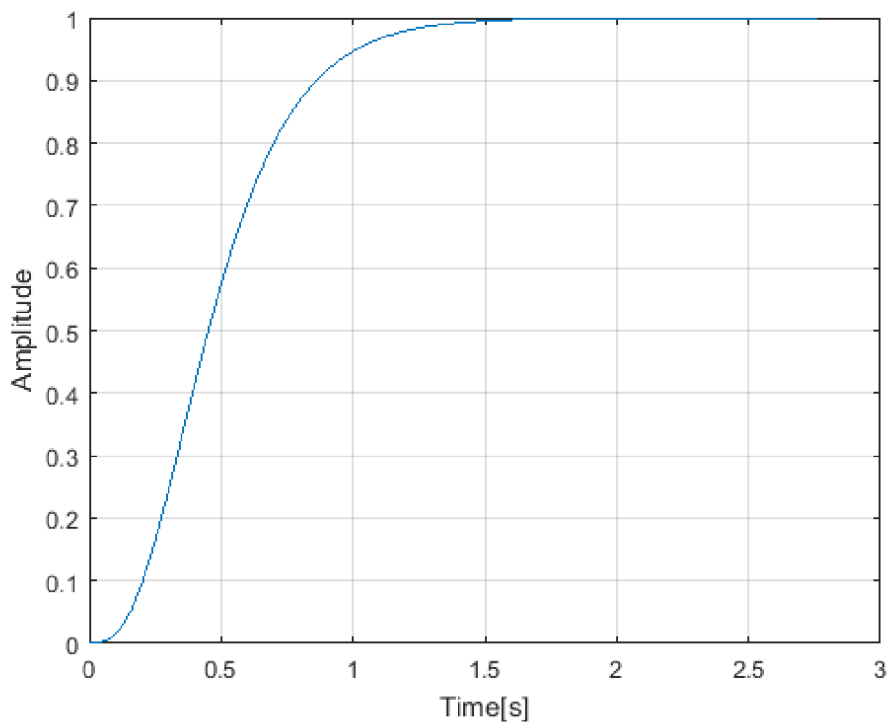
$$\dot{\mathbf{x}} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 4,5746 \\ 0 & 0 & -5,1677 * 10^{-9} \end{bmatrix} \mathbf{x} + \begin{bmatrix} 0 \\ 0 \\ 5,1677 * 10^{-9} \end{bmatrix} \mathbf{u}_\alpha \quad (61)$$

$$\mathbf{y}_x = [1 \quad 0 \quad 0] \mathbf{x} \quad (62)$$

Hodoty \mathbf{R}_i a \mathbf{r}_i vypočtené programem jsou:

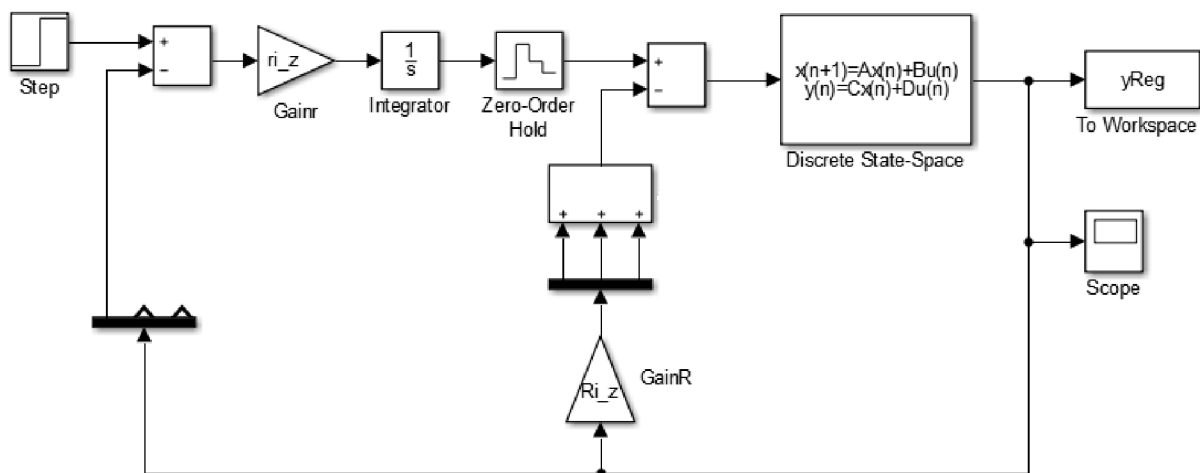
$$\mathbf{R}_i = [10,3554 \quad 1,9595 \quad -0,2567]$$

$$\mathbf{r}_i = 19,8977$$



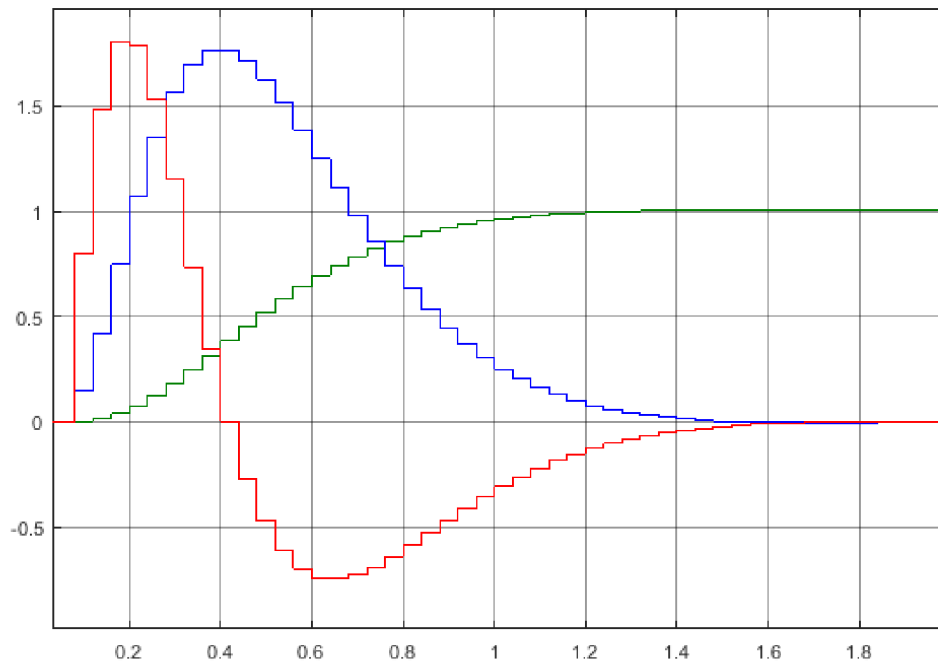
Obr. 24: Odezva systému na jednotkový skok

Dál byl vytvořen model v Simulinku, viz obrázek 25.



Obr. 25: Model systému se stavovým regulátorem

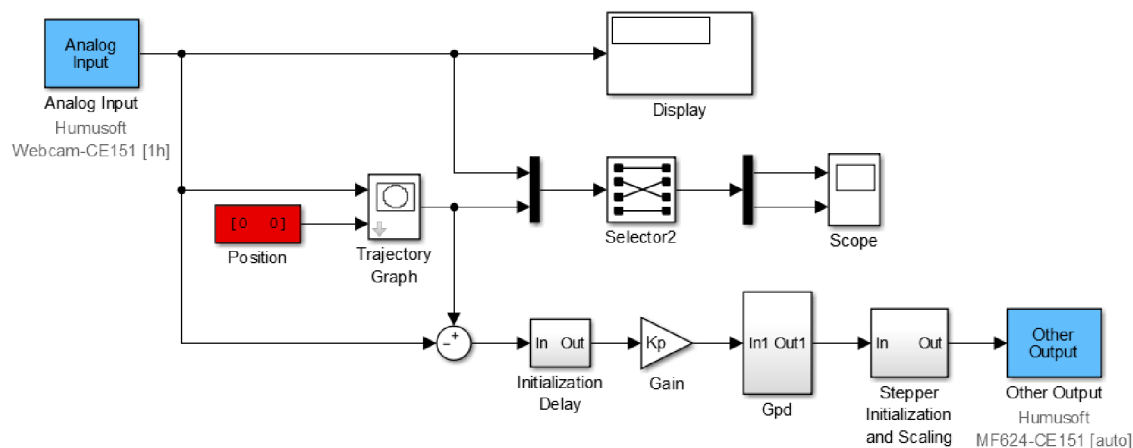
Na grafu 26 získaném z bloku "Scope" z modelu v Simulinku vidíme identickou odezvu systému na jednotkový skok.



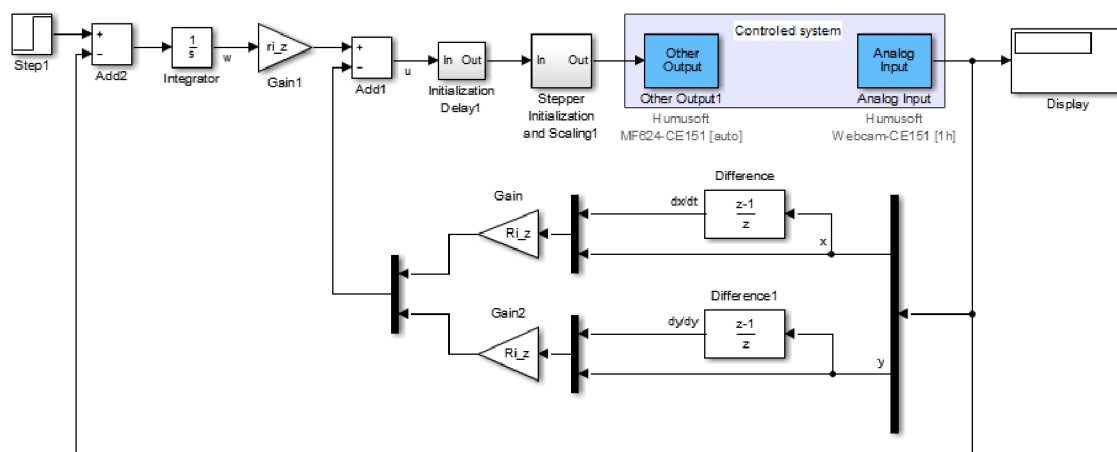
Obr. 26: Odezva regulovaného systému na skok požadované hodnoty (tři stavy, výstup koresponduje se stavem vyobrazeným zeleně)

3.4 Výsledky experimentu na reálném modelu

Pro testování řídicích algoritmů na reálném modelu byla změněna počáteční schéma řízení v Simulinku, ale bohužel žádný z vypočtených regulátorů nedal požadovaný výsledek. Možné příčiny neúspěchu zahrnují výpočetní omezení osobního počítače použitého pro spuštění prostředí Simulink, pomalé získávání informací z kamery či vliv odstředivých zrychlení, jež linearizovaný model nedokáže pochytit. Nejpravděpodobnější příčinou nestability však představuje fakt, že se ze servopohonů na řízenou desku, a dále i na míček, přenáší intenzivní vibrace, jež způsobují ztrátu kontaktu míče s deskou. V těchto neustále se vyskytujících okamžicích zcela přestává platit odvozený model dynamiky systému. Možným řešením bylo získání přímého přístupu k hardwarovému a elektronickému vybavení servosystému nebo změna softwaru, což nebylo cílem této práce. Proto pro vizualizaci výsledků praktické části práce bylo v Simulinku vytvořeno 3D model systému pomocí knihovny Simscape. Popis schématu je uveden v následující části.



Obr. 27: Upravené schéma pro PD regulátor s filtrem



Obr. 28: Upravené schéma pro zpětnovazební regulátor s integrátorem

3.5 Počítačová simulace

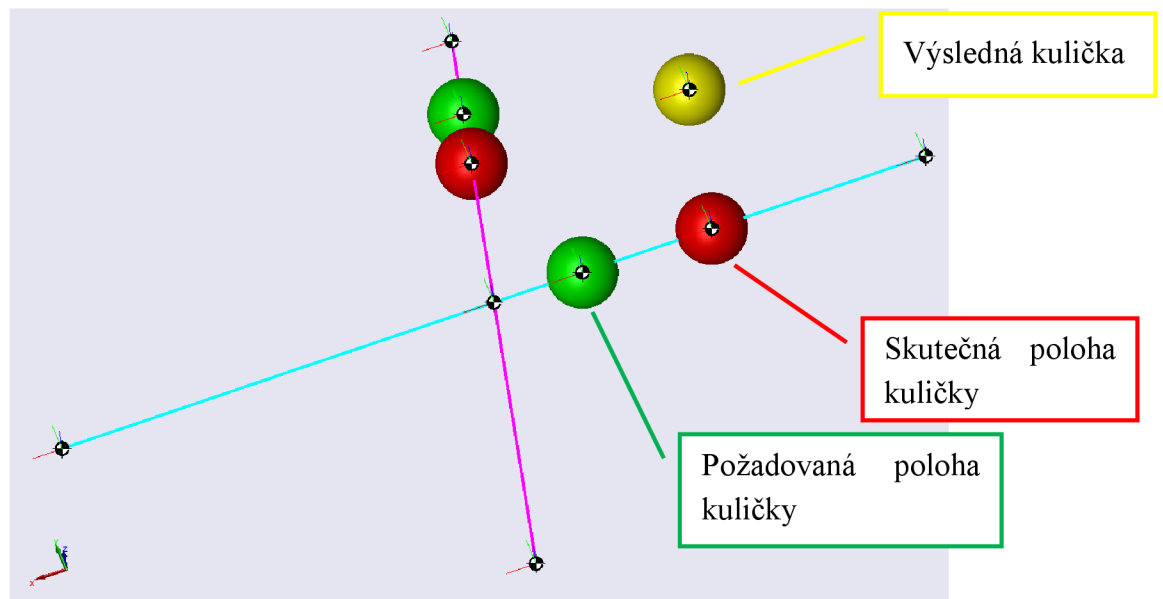
3.5.1 Konstrukce 3D modelu

Pro vytvoření 3D modelu byly použity bloky první generace knihovny Simscape. Protože reálný model lze považovat za systém ze dvou nezávislých os, byl pro začátek vytvořen model systému v jedné ose. Systém se skládá ze dvou těles propojených spojem „Weld“. Také v tomto místě je blok „Revolute“ spojující desku se zemí a zajišťující její rotační pohyb podél jedné osy. Dále pomocí spojení „Prismatic“ kulička byla umístěna na rovinu a byla nastavena osa pro její pohyb. Simulován byl také vliv tření na provoz systému.

Dalším krokem bylo provedení identifikace simulovaného 3D modelu systému, sestavení jeho matematického modelu a výpočet regulátoru. Dále bylo vyvinuto schéma pro určení a zobrazení požadované polohy a schéma pro simulaci odrazu kuličky od boční stěny desky.

Dalším krokem byl vývoj identického schématu pro druhou osu, jeho identifikace a regulace. Dále dvě osy byly spojeny. Ve výsledku dostáváme dvě nezávisle pohyblivé desky a dvě nezávislé kuličky ve dvou osách. Výsledná kulička je superpozice těchto dvou kuliček.

Obrázek 29 ukazuje výsledný model. Osa x je označena modrou, růžovou – osa y . Zelené kuličky označují požadovanou pozici, červené označují skutečnou pozici. Výsledkem pohybu dvou červených kuliček je žlutá koule, která simuluje pohyb skutečné koule.



Obr. 29: Model systému v Simulinku s využitím SimScape knihovny

3.5.2 Identifikace a regulace modelu

Pro identifikaci a návrh diskrétního stavového regulátoru se sumátorem byl použit upravený program z diplomové práce Bc. Zdeňka Cejпка [12].

Chování a vlastnosti navrženého 3D modelu (Simscape) umožňují takové akce, které by odpovídaly studiu a analýze vlastností provedených na reálném modelu. Příkladem je provedená identifikace soustavy. Kulička je umístěna ve středu vodorovné pevné desky, na systém působí vstupní veličina (volen jednotkový skok), která způsobí naklonění desky. Vlivem dosaženého náklonu se dá kulička do pohybu. Deska se považuje za nekonečnou rovinu s libovolnou definicí sklonu, přičemž je vždy zachován styk míčku a roviny. Na schématu se pomocí bloků "Joint Sensor" měří vstupní napětí, poloha kuličky a úhel náklonu desky. Pomocí diferenciací byly získány rychlosti kuličky a desky. Blok "To Workspace" odstarává přenos těchto dat do *workspace* paměťového prostoru Matlabu, pro jejich další zpracování užitým programem. Systém v Simulinku pracuje jako spojitý, ale skutečný systém je diskrétní proto před blokem "To Workspace" je umístěn blok "Zero Order Hold", který diskretizuje signál.

Blokem “To Workspace“ se do programu přenáší vstupní napětí, poloha kuličky, úhel náklonu desky a rychlosti kuličky a desky. Z těchto proměnných je složena matice **L**, která v sobě zahrnuje matici **A** a **B**. Aby bylo možné stanovit vstupní matici **B**, je třeba na soustavu působit nenulovou hodnotou vstupního signálu. Jeho absence povede na nulový determinant matice **L**. Matice **A** představuje stavy systému. Za stavy jsou považovány poloha kuličky a úhel náklonu desky a jejich diference.

Identifikace se provádí metodou nejmenších čtverců. Výsledky jsou vykresleny do čtyř grafů. Z těchto grafů můžeme vyhodnotit, jestli identifikovaný model dostatečně věrně popisuje chování měřeného systému. Během identifikace se mohou projevit kumulativní chyby, jež nejsou žádným způsobem kompenzovány.

Pro regulaci byl použit diskretní stavový zpětnovazební regulátor se sumátorem. Vypočtené hodnoty \mathbf{R}_i a \mathbf{r}_i pro osu x jsou následující:

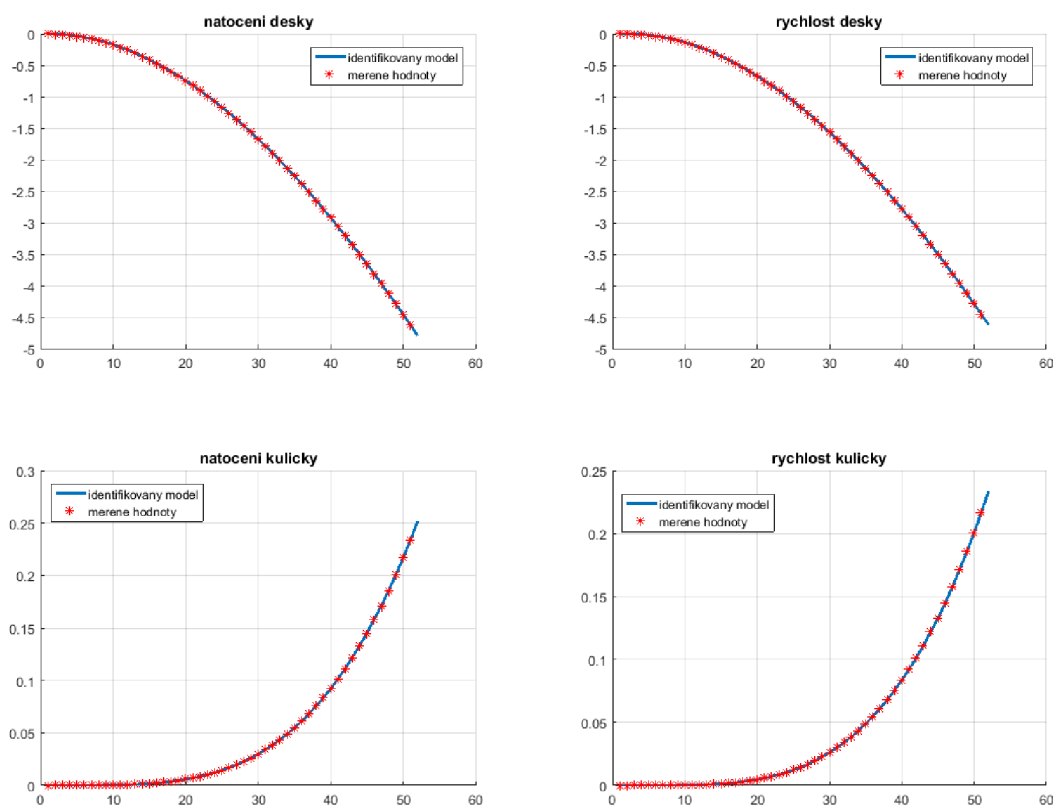
$$\mathbf{R}_i = [26,8 \quad -16,6 \quad -9702,7 \quad 8436,1]$$

$$\mathbf{r}_i = 65,8$$

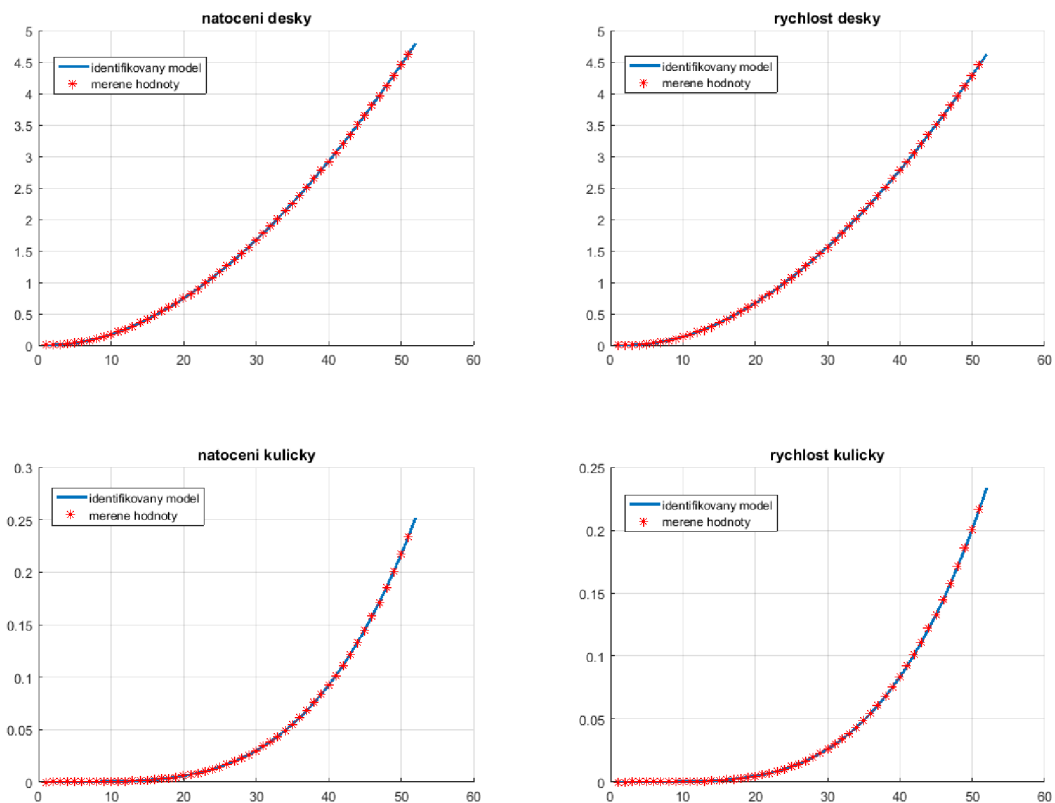
Vypočtené hodnoty \mathbf{R}_i a \mathbf{r}_i pro osu y jsou následující:

$$\mathbf{R}_i = [26,8 \quad -16,6 \quad 9702,7 \quad -8436,1]$$

$$\mathbf{r}_i = -65,8$$



Obr. 30: Identifikace systému 3D modelu v ose x

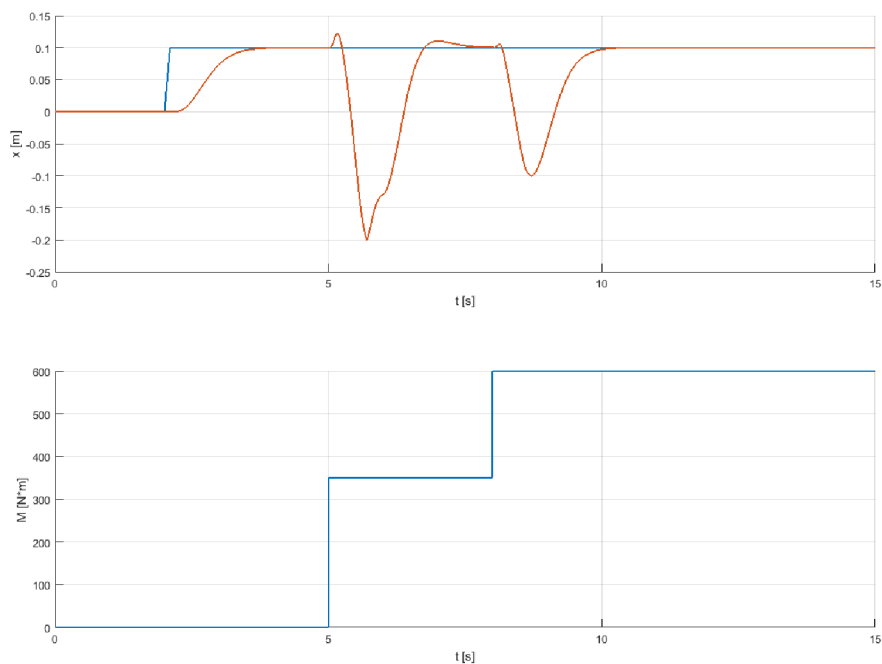


Obr. 31: Identifikace systému 3D modelu v ose y

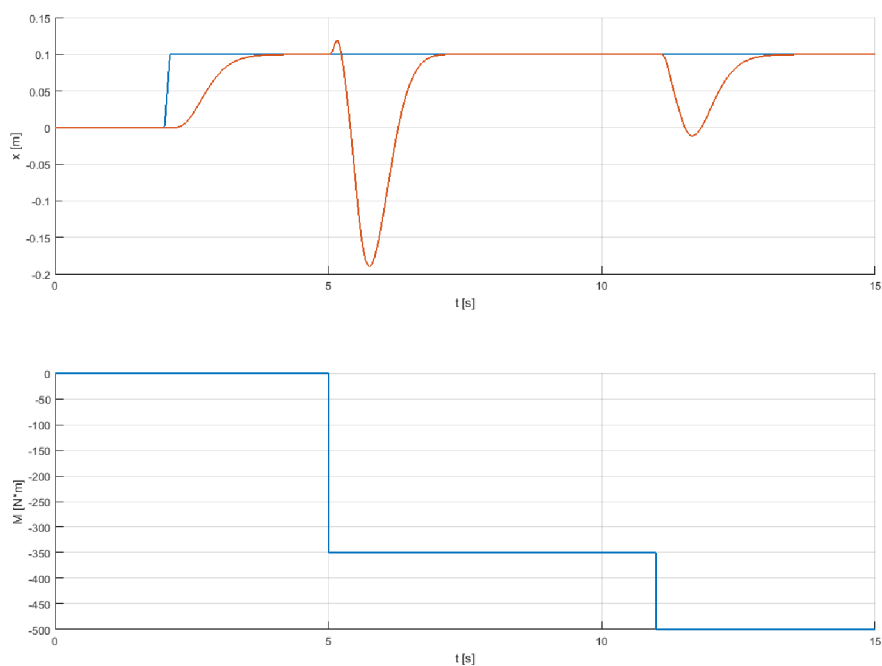
3.5.3 Výsledky simulací 3D modelu

Během verifikace konstruovaného modelu byly provedeny různé experimenty, při kterých byla testována schopnost řízení sledovat požadovanou pozici kuličky, odezva systému na skok poruchové veličiny různých velikostí. Dále byl proveden test průběhu simulace v okamžiku odrazu koule od imaginární boční stěny desky. Následující obrázky 32 a 33 ukazují grafy reakce systému v obou osách na níže uvedené akce. V horním grafu je modrou křivkou znázorněn průběh požadované polohy, průběh skutečné polohy je znázorněn červeně. Průběh poruchové veličiny je zachycen ve spodním grafu. V čase 2 [s] byla skokově změněna požadovaná poloha koule. Graf ukazuje, že kulička dosáhne požadované polohy v čase 4 [s]. V okamžiku 5 [s] byl současně vyslán skokový signál různé velikosti do obou os. Kulička se při tomto experimentu v čase 6 [s] odrazí od boční stěny desky. Doba ustálení je 3 [s], přičemž překmit dosáhl 10 [%]. Dále, v časech 8 [s], resp. 11 [s], byl vyslán skokový signál do osy x , resp. osy y . Doba přechodu při této excitaci byla 3 [s], s minimálním překmitem, viz obrázek 32 a 33.

Změna poruchové veličiny je volena nesmyslně velká, aby se viditelně projevila. Rozumné velikosti poruchové veličiny pro realizovatelný systém mohou nabývat jen řádově menších hodnot.

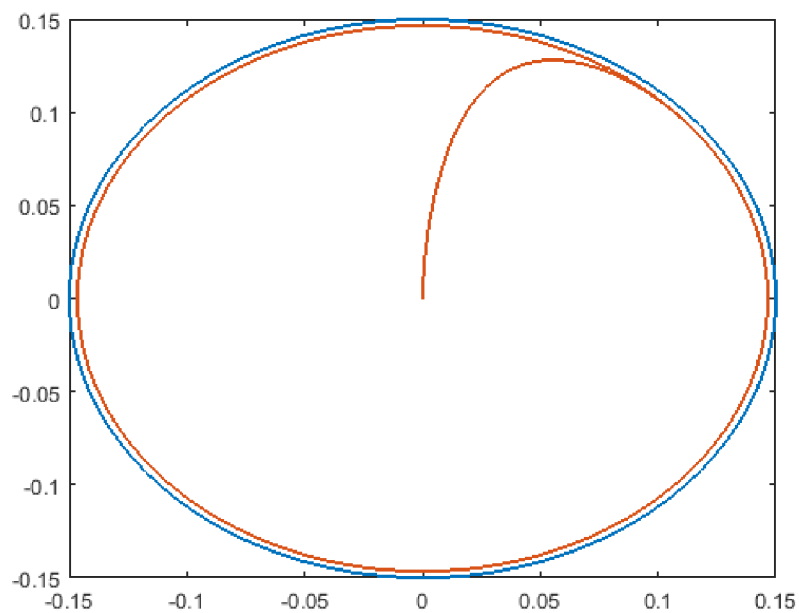


Obr. 32: Odezva systému 3D modelu na změnu požadované polohy a změnu poruchové veličiny v ose x

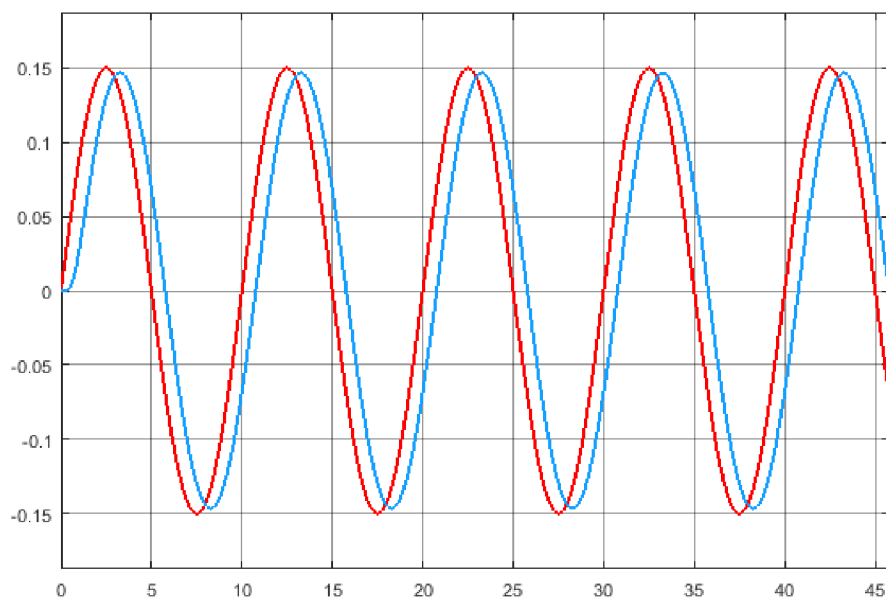


Obr. 33: Odezva systému 3D modelu na změnu požadované polohy a změnu poruchové veličiny v ose y

Při nastavení řídicího signálu sinusového průběhu (nastavení požadované polohy) pro obě osy, se podařilo posouvat kuličku po kružnici, jak ukazuje obrázek 33.



Obr. 34 Požadovaný a skutečný pohyb kuličky po kružnici



Obr. 35: Sledování požadovaného a skutečného signálu v jedné ose

4 ZÁVĚR

V této diplomové práci byl představen model CE 151 “*Ball & Plate*” firmy Humusoft, tedy jeho struktura, parametry, matematický popis, návrh vhodného regulátoru a jeho 3D simulační model. Skutečný model nebyl ideální, především kvůli fyzické implementaci servosystému a technickým vlastnostem použité kamery. Možným řešením těchto obtíží by bylo získání přímého přístupu k hardwarovému a elektronickému vybavení nebo ke změně softwaru, což nebylo cílem této práce.

Na začátku práce jsem se pozorně seznámila s manuálem, konstrukcí samotného modelu a softwarem dodaným výrobcem. Následovaly první jednoduché experimenty s modelem a první pokusy identifikovat model. Dále bylo provedeno měření odezvy modelu podél obou os pro čtyři úhly náklonu. Vyhodnocením naměřených dat v Matlabu byly získány přenosové funkce systému, které se liší od funkcí uvedených v manuálu.

V teoretické části práce je uveden popis vybraných metod návrhu regulátoru, tedy metody kořenového hodografu a metody umisťování pólů ve stavovém prostoru. Pomocí metody kořenového hodografu byl navržen PD regulátor, pro návrh stavového regulátoru byla použita struktura regulátoru s integrátorem. V počítačové simulaci vykazovaly obě metody velmi dobré výsledky. Aplikace těchto metod na skutečný model takový úspěch nepřinesla.

Další fází práce byl vývoj 3D modelu v prostředí Simulink. Byly použity bloky první generace knihovny Simscape-Simmechanics. Výsledný model se skládá ze dvou nezávislých systémů pro každou osu. Aby se model přiblížil realitě, je simulováno viskózní tření a odraz koule od boční stěny desky. Z dat naměřených na simulovaném modelu byl vytvořen lineární model systému ve stavovém prostoru. Pro regulaci byl použit zpětnovazební regulátor s integrátorem. Vytvořený model umožňuje provádět identifikaci, měnit metody řízení a provádět experimenty jako na reálném modelu. Předností tohoto 3D modelu je názornost, dostupnost a zrealizovaná ukázková řešení návrhu regulátoru.

Pro všechny výpočty a simulace v této práci byl využit MATLAB/Simulink, jeho toolboxy a knihovny.

SEZNAM POUŽITÉ LITERATURY

- [1] Kirchner, Miroslav. *Laboratorní model CE 151 - kulička na ploše*. Zlín, 2011. Diplomová práce. UTB ve Zlíně. Vedoucí práce Ing. František Gazdoš, Ph.D.
- [2] Luboš, Spaček. *Digital Control of CE 151 Ball & Plate Model*. Zlín, 2016. Diplomová práce. UTB ve Zlíně. Vedoucí práce prof. Ing. Vladimír Bobál, CSc.
- [3] Debre, Michal. *Identifikace a návrh řízení pro model kulička na ploše*. Praha, 2010. Bakalářská práce. ČVUT v Praze. Vedoucí práce Ing. Jiří Řehoř.
- [4] Míková, Jitka. *Aplikace vybraných algoritmů řízení na model CE 151 - kulička na ploše*. Zlín, 2012. Diplomová práce. UTB ve Zlíně. Vedoucí práce Ing. František Gazdoš, Ph.D.
- [5] Fan, X.; Zhang, N.; Teng, S.: *Trajectory planning and tracking of ball and plate system using hierarchical fuzzy control scheme*. Fuzzy Sets and Systems, ročník 144, č. 2, 2004: s. 297 – 312, ISSN 0165-0114, doi:[https://doi.org/10.1016/S0165-0114\(03\)00135-0](https://doi.org/10.1016/S0165-0114(03)00135-0). URL <http://www.sciencedirect.com/science/article/pii/S0165011403001350>
- [6] František Dušek, Daniel Honc, Rahul Sharma K. “*Modelling of Ball and Plate System Based on First Principle Model and Optimal Control*“. Štrbské Pleso, Slovakia, 2017.
- [7] Matej Oravec, Anna Jadlovská. “*Optimal control of the mechatronical laboratory model B&P_KYB*“. Košice, 2014. Fakulta elektrotechniky a informatiky, Technická univerzita v Košicích, Slovenská republika. ISBN 978-80-553-1704-5
- [8] CE 151 kulička na ploše. *HUMUSOFT s.r.o.* [online]. 2000 [cit. 2012-04-19]. Dostupné z: <http://www.humusoft.cz/produkty/models/ce151/>
- [9] HUMUSOFT. *CE 151 Ball & Plate Apparatus User's Manual*. Prague, 2006.
- [10] Ivan Švarc, Radomil Matoušek, Miloš Šeda, Miluše Vitečková. *Automatické řízení*. Brno: ACADEMICKÉ NAKLADATELSTVÍ CERM, 2011. ISBN 978-80-214-4398-3.
- [11] Tomáš, Březina. *Stavový zpětnovazební regulátor s integrátorem*. 2017.
- [12] Bc. Zdeňek Cejpek. *Návrh řízení rotačního inverzního kyvadla*. Brno, 2019. Diplomová práce. Vysoké učení technické v Brně, Fakulta strojního inženýrství, Ústav automatizace a informatiky. Vedoucí práce doc. Ing. Radomil Matoušek, Ph.D.
- [13] Katsuhiko, Ogata. *Modern Control Engineering (5th Edition)*. Pearson, 2009. ISBN-13: 978-0136156734.
- [14] Karban, Pavel. *Výpočty a simulace v programech Matlab a Simulink*. Praha: BEN technická literatura, 2007.220 s.ISBN 978-80-251-1448-3.

SEZNAM OBRÁZKŮ

Obr. 1: Schéma modelu B&P (Ball & Plate) [8].....	19
Obr. 2: GUI dodávaného demo programu pro model B&P.....	21
Obr. 3: Základní Simulink/Matlab model B&P (pro CE 151).....	22
Obr. 4: Schéma servosystému [9]	27
Obr. 5: Blokový diagram servosystému [9].....	27
Obr. 6: Blokové schéma dynamiky nelineárního servosystému [9]	28
Obr. 7: Bloková schéma systému ve stavovém prostoru.....	29
Obr. 8: Schéma stavového zpětnovazebního regulátoru s integrátorem[11].....	31
Obr. 9: Pohled na model a obraz z kamery.....	35
Obr. 10: Schéma převodu napětí na úhel.....	36
Obr. 11: Závislost náklonu α [rad] na změně vstupního napětí u [MU] pro osu x	37
Obr. 12: Aproximovaný průběh polohy míčku pro náklon 0,8 MU v ose x	39
Obr. 13: Aproximovaný průběh polohy míčku pro náklon 0,6 MU v ose x	39
Obr. 14: Aproximovaný průběh polohy míčku pro náklon 0,4 MU v ose x	40
Obr. 15: Aproximovaný průběh polohy míčku pro náklon 0,2 MU v ose x	40
Obr. 16: Aproximovaný průběh pro náklon 0,8 MU v ose y	41
Obr. 17: Aproximovaný průběh pro náklon 0,6 MU v ose y	42
Obr. 18: Aproximovaný průběh pro náklon 0,4 MU v ose y	42
Obr. 19: Aproximovaný průběh pro náklon 0,2 MU v ose y	43
Obr. 20: Grafy neregulovaného systému pro osu x	44
Obr. 21: Grafy pro osu x s PD regulátorem.....	45
Obr. 22: Grafy pro osu y s PD regulátorem.....	46
Obr. 23: Grafy pro osu x s PD regulátorem s filtrací	47
Obr. 24: Odezva systému na jednotkový skok.....	48
Obr. 25: Model systému se stavovým regulátorem.....	48
Obr. 26: Odezva regulovaného systému na skok požadované hodnoty	49
Obr. 27: Upravené schéma pro PD regulátor s filtrem.....	50
Obr. 28: Upravené schéma pro zpětnovazební regulátor s integrátorem	50
Obr. 29: Model systému v Simulinku s využitím SimScape knihovny	51
Obr. 30: Identifikace systému 3D modelu v ose x	52
Obr. 31: Identifikace systému 3D modelu v ose y	53
Obr. 32: Odezva systému 3D modelu na změnu požadované polohy.....	54
Obr. 33: Odezva systému 3D modelu na změnu požadované polohy.....	54
Obr. 34 Požadovaný a skutečný pohyb kuličky po kružnici	55
Obr. 35: Sledování požadovaného a skutečného signálu v jedné ose	55

SEZNAM TABULEK

Tab. 1: Simulované hodnoty.....	36
Tab. 2: Naměřené hodnoty	37

SEZNAM ZKRATEK

GUI – graphical user interface,

LQ – linear quadratic,

PC – personal computer,

PID – regulátor, složený z proporcionální, integrační a derivační části,

TTL – transistor-transistor logic,

MU – machine units,

MIMO – multiple input multiple output.

PŘÍLOHY

Skript použitý pro identifikaci soustavy CE151

```

clear all;
close all;
clc
global t A u y

load ('y.mat');

t=y(:,1);
t=t-min(t);
u=y(:,3);
y=y(:,2);
y=y-min(y);

%% struktura obrazoveho prenosu
disp('Struktura obrazoveho prenosu')
disp('          x(1)          K ')
disp('F(s)= ----- = -----')
disp('      (x(2).s + 1)s^2      (Ts + 1)s^2')
T = 0.2; %pocatecni odhad casove konstanty
K = 5; %pocatecni odhad zesileni
A = [1 0]; %polynom A = jmenovatel
F = tf([K],[conv([conv(A,A)],[T 1])]) %LTI objekt prenosove funkce
x = [K T]; %vektor hledanych parametru x

%% hledani optimalnich parametru - minimalizace J
x = fminsearch('crit', x);

%% vystupy
disp('Optimalizovany vektor x:')
x
disp('Konecna hodnota kriteria J:')
critT3(x)
Kid = x(1); %vypoctene zesileni
Aid = [conv([conv([1 0],[1 0]],[x(2) 1])]);
disp('Nalezeny obrazovy prenos:')
Fid = tf(Kid, Aid)
disp('Zesileni systemu:')
Kid
disp('Koreny charakteristicke rovnice:')
roots(Aid)
disp('Casove konstanty ve jmenovateli:')
1./roots(Aid)
[yi,ti] = lsim(Fid, u, t); %reakce nalezeneho systemu na puvodni
prirustkovy vstup
figure; plot(t, y, t, u, ti, yi) %porovnani odezvy mereneho systemu a
nalezeneho obrazoveho prenosu
grid on
xlabel('t[s]'); ylabel('y[-]');
legend('Namereno', 'Vstupni napeti', 'Aproximovano');

```

Skript "critT.m" definující kritériální funkci pro potřeby identifikace modelu soustavy CE151

```
function J = critT(x)
global t y u A

sys = tf(x(1), [conv([conv([1 0], [1 0]), [x(2) 1])]); %obrazovy prenos
[yi, ti] = lsim(sys, u, t); %vypocet odezvy systemu na buzeni
J = sum((y - yi).*(y - yi)); %minimalizovana funkce J
```

Skript použitý pro návrh regulátoru metodou stavového prostoru

```
clear all;
clc;

K = 4.5746;
Tm = 1.9351*10^(-8);
Ts = 0.04;

A = [ 0 1 0;
      0 0 K;
      0 0 -1/Tm];
B = [0; 0; 1/Tm];
C = eye(size(A,1));
D = zeros(size(A,1),1);
sys = ss(A,B,C,D);
sys_d = c2d(sys, Ts);
pole(sys_d)
Ad = sys_d.A;
Bd = sys_d.B;
Cd = sys_d.C;
Dd = sys_d.D;

n = size(A, 1);
X = [A, zeros(n,1); -C(1,:), 0];
Y = [B(:,1); -D(1,:)];
c = 10*[-1 -1.1 -0.5 -0.8];

sysxy = c2d(ss(X, Y, eye(1,4), 0), Ts);
X_z = sysxy.A;
Y_z = sysxy.B;
c_z = exp(c*Ts);
Z_z = place(X_z, Y_z, c_z);
Ri_z = Z_z(1:n)
ri_z = -Z_z(end)

A2 = X_z - Y_z * Z_z;
B2 = Y_z;
C2 = eye(n+1)
D2 = zeros(n+1,1);

sys1 = ss(A2, B2, C2, D2, Ts);
sys1.A = A2;
sys1.B = B2;
sys1.C = C2;
sys1.D = D2;
```



```

figure(1)
dcgainR = dcgain(sys1);
stepN1 = step(sys1)/dcgainR(n+1);
plot(0:Ts:(length(stepN1)-1)*Ts, stepN1(:,4))
grid on
xlabel('Time[s]'); ylabel('Amplitude');

```

Skript použitý pro identifikaci a regulaci 3D modelu v Simulinku

```

clear all
close all
clc

% Parameters for simulation
Ts = 0.04;
l = 0.4;
mKul = 0.001;
mTyci = 0.01;
bKul = 0.0000000005;
bTyce = 0.01;

%% V OSE X
% simulation itself, produces xMes and uMes
tSim = 50*Ts;
sim('SimIdentX.slx')

% Identification - getting the parameters
Lx = [xMes(2:end-1,:), uxMes(2:end-1,:)];

Aprocx = Lx\xMes(3:end,:);

Bx = Aprocx(end,:)' ;
Ax = Aprocx(1:end-1,:);
nx = size(Ax,1);
Ax = Ax';
Cx = eye(nx);
Dx = zeros(nx,1);

xId = zeros(size(xMes,2),nx);
xId(2,:) = xMes(2,:);
for i = 2:size(xMes,1)
    xId(i+1,:) = Ax*xId(i,:)' + Bx*uxMes(i);
end

%plotting
figure(1)
subplot(2,2,1)
hold on
plot(xId(:,1), 'LineWidth',2);
plot(xMes(:,1), 'r*')
setLegend();
title('natoceni desky');
grid on

subplot(2,2,2)
hold on
plot(xId(:,2), 'LineWidth',2);
plot(xMes(:,2), 'r*')

```

```

setLegend();
title('rychlost desky');
grid on

subplot(2,2,3)
hold on
plot(xId(:,3), 'LineWidth',2);
plot(xMes(:,3) ', 'r*')
setLegend();
title('natoceni kulicky');
grid on

subplot(2,2,4)
hold on
plot(xId(:,4), 'LineWidth',2);
plot(xMes(:,4) ', 'r*')
setLegend();
title('rychlost kulicky');
grid on

% Regulation
nx = size(Ax,1);
Xx = [Ax zeros(nx,1); ...
-Cx(3,:), 1];

Yx = [Bx(:,1); ...
0];

evRx = 0.1*[-10 -11 -12 -13];
evdRx = exp(evRx*Ts);

evZx = 1*[-5 -6 -7 -8 -9];
evdZx = exp(evZx*Ts);

Rx = acker(Ax, Bx, evdRx)
Zx = acker(Xx, Yx, evdZx)

%% V OSE Z
% simulation itself, produces zMes and uzMes
tSim = 50*Ts;
sim('SimIdentZ.slx')

% Identification - getting the parameters
Lz = [zMes(2:end-1,:), uzMes(2:end-1,:)];

Aprocz = Lz\zMes(3:end,:);

Bz = Aprocz(end,:)';
Az = Aprocz(1:end-1,:);
nz = size(Az,1);
Az = Az';
Cz = eye(nz);
Dz = zeros(nz,1);

zId = zeros(size(zMes,2),nz);
zId(2,:) = zMes(2,:);
for i = 2:size(zMes,1)
    zId(i+1,:) = Az*zId(i,:) + Bz*uzMes(i); end

```

```
%plotting
figure(2)
subplot(2,2,1)
hold on
plot(zId(:,1), 'LineWidth',2);
plot(zMes(:,1), 'r*')
setLegend();
title('natoceni desky');
grid on

subplot(2,2,2)
hold on
plot(zId(:,2), 'LineWidth',2);
plot(zMes(:,2), 'r*')
setLegend();
title('rychlost desky');
grid on

subplot(2,2,3)
hold on
plot(zId(:,3), 'LineWidth',2);
plot(zMes(:,3), 'r*')
setLegend();
title('natoceni kulicky');
grid on

subplot(2,2,4)
hold on
plot(zId(:,4), 'LineWidth',2);
plot(zMes(:,4), 'r*')
setLegend();
title('rychlost kulicky');
grid on

% Regulation
nz = size(Az,1);
Xz = [Az zeros(nz,1); ...
-Cz(3,:), 1];

Yz = [Bz(:,1); ...
0];

evRz = 0.1*[-10 -11 -12 -13];
evdRz = exp(evRz*Ts);

evZz = 1*[-5 -6 -7 -8 -9];
evdZz = exp(evZz*Ts);

Rz = acker(Az, Bz, evdRz)
Zz = acker(Xz, Yz, evdZz)

tSim = 15;
```