

České vysoké učení technické v Praze  
Fakulta elektrotechnická  
Katedra řízení



Bakalářská práce

## **Identifikace a návrh řízení pro model kulička na ploše**

*Michal Debre*

Vedúci práce: Ing. Jiří Řehoř

Študijný program: Elektrotechnika a informatika, dobíhající, Bakalářský

Obor: Kybernetika a měření

10. ledna 2010



## Poděkování

Moja veľká vďaka patrí viacerým ľuďom. Najdôležitejšiu rolu zohral pri tvorení tejto práce pán Ing. Jiří Řehoř, vedúci. Za jeho ochotu stretávať sa a prejednávať jednotlivé problémy a pasáže mu veľmi pekne ďakujem. Ďalej by som chcel poďakovať mojej mamke, bratovi a jeho rodinke. Podporovali ma a verili mi, že to dokážem. Nemalú zásluhu majú aj moji kamaráti, ktorí mi napovedali ako postupovať, keď som sa dostal do slepej uličky. Rád by som uviedol ich mená, Bc. Ján Michalko, Bc. Matej Pčolka a Bc. Eva Žáčeková. Ďakujem Vám.



# Prohlášení

Prohlašuji, že jsem svou bakalářskou práci vypracoval samostatně a použil jsem pouze podklady (literaturu, projekty, SW atd.) uvedené v příloženém seznamu.

V Praze dne 7. 1. 2010

.....



# Abstract

Bachelory thesis Modeling and Control design for the ball and plate system is using modern theories of controlling. In the first part of this thesis we'll create theoretical model and consequently we'll identify controlling system. For identification we use GREY BOX method. The ball and plate system will be controlled using two regulator types LQ or PID. Regulators will be compared and evaluated.

# Abstrakt

Bakalárska práca identifikácie a návrh riadenia pre model kulička na ploše využíva moderné teórie riadenia. V prvej časti tejto práce zostrojíme teoretický model a následne budeme identifikovať systém a riadiť ho. Pre identifikáciu použijeme metódu GREY BOX. Kulička na rovine bude riadená dvomi typmi regulátorov LQ alebo PID. Regulátory budú porovnané a zhodnotené.





# Obsah

<b>1</b>	<b>Úvod</b>	<b>1</b>
<b>2</b>	<b>Identifikácia</b>	<b>3</b>
2.1	Model	3
2.2	Analytické modelovanie(white box)	4
2.2.1	Príklad	4
2.3	Black-box	5
2.4	Grey-box	6
2.4.1	Úvod	6
2.4.2	Teoretický základ	6
2.5	Najmenšie štvorce	7
2.5.1	Riešenie LS pomocou pseudoinverzie	9
2.5.2	Príklad	10
<b>3</b>	<b>Grey-Box</b>	<b>13</b>
3.1	Optimalizácia	13
3.1.1	Optimalizačný solver	14
3.1.2	Vstupné argumenty	14
3.1.3	Možnosti	15
3.2	Popis funkcie prederrSISO	15
3.2.1	Volanie funkcie prederrSISO	16
3.2.2	Validace modelu	16
3.3	Príklad	17
<b>4</b>	<b>Popis systému gulička na rovine</b>	<b>19</b>
4.1	Teoretický model gulička na rovine	19
4.2	Nelineárny model	20
4.3	Linearizácia a stavový popis	22
4.4	Popis riadiacej karty humusoft MF 624	24
4.5	Servo systém	24
4.6	Identifikácia systému gulička na naklonenej rovine	25
<b>5</b>	<b>Návrh riadenia</b>	<b>29</b>
5.1	PID regulátor	29
5.2	LQR (Lineárny kvadratický regulátor)	31

<b>6 Závěr</b>	<b>33</b>
<b>Literatura</b>	<b>35</b>
<b>A lsqnonlin solver</b>	<b>37</b>
A.0.1 Výstupné argumenty . . . . .	37
<b>B Obrazová příloha</b>	<b>39</b>
<b>C CD</b>	<b>41</b>

# Seznam obrázků

1.1	Wattov regulátor	1
2.1	Matematické kyvadlo	5
2.2	Black box	6
2.3	Najmenšie štvorce - príklad	11
3.1	Zle identifikované parametre	18
3.2	Správne identifikované parametre	18
4.1	Nelineárny model	22
4.2	Lineárny model	23
4.3	Odozva na skok	25
4.4	Vstup, výstup	27
4.5	Identifikácia systému 1	27
4.6	Vstup, výstup	28
4.7	Identifikácia systému 2	28
5.1	PSD regulátor zapojený na systém č.1	30
5.2	PSD regulátor zapojený na systém č.2	30
5.3	Výstupy	32
5.4	LQ regulátor	32
B.1	Simulinková schéma zapojenia PSD do systému	39
B.2	PSD regulátor	40
B.3	Zapojenie LQ regulátora	40

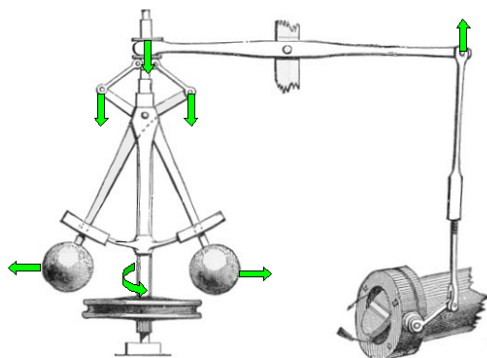


# Kapitola 1

## Úvod

Riadenie je technológia, ktorú ľudstvo využíva tisíročia. Prvé zmienky riadiacich systémov sahajú až do praveku 2000 rokov pred našim letopočtom. Polynézania v tej dobe začali podnikáť dlhé plavby v Pacifiku. Ich jednoduché lode vydlabané len z kmeňa stromu boli veľmi vratké. Potrebovali teda vymyslieť niečo jednoduché, nejaký kvázy regulátor, ktorý by im pomohol potlačiť aspoň čiastočne vplyv vetra, vln. Riešenie sa naskytlo v podobe váhadla. Takto vznikli prvé vahadlové kanoe a katamarány [16].

Snáď za jeden z najkrajších a najznámejších regulátor môžeme považovať Wattov odstredivý regulátor, ktorý sa používal ku stabilizácii otáčok parného stroja. Bol zavedený Jamesom Wattom v roku 1782. Skladá sa z dvoch závaží, ktoré rotujú a sú poháňané strojom, ktorého otáčky majú byť regulované. Čím rýchlejšie tieto závažia rotujú, tým väčšia je výchylka od zvislej osy rotácie. Uvedené vychýlenie je nad ich ukotvením prevedené na zvislý pohyb, ktorý je ďalej pákou a ťahlom prevedené k ventilu privádzajúcemu paru do stroja. Je tak realizovaná mechanická záporná spätná väzba, ktorá dovoľuje pôsobením pomerne malých síl regulovať veľmi výkonný stroj [4].



Obrázek 1.1: Wattov regulátor

Za posledných pár desaťročí sa zvýšili nároky na riadenie. Vytvárajú sa mnohé nestabilné systémy, ktoré potrebujú stabilizovať. Bez stabilizovaného riadenia by teda nemohli

existovať niektoré dôležité procesy. Veľmi zaujímavým príkladom je družica na obežnej dráhe Zeme, ktorá predstavuje veľmi nestabilný a ťažko vybuditeľný systém. Je veľmi obtiažne s ňou manévrovať. Keby sme pre ňu nevedeli navrhnuť stabilizované riadenie, ľahko by sa nám mohlo stať, že pri pokuse o pohyb by sa družica začala nekontrolovateľne otáčať a mohla by aj opustiť obežnú dráhu okolo Zeme.

Pri moderných metódach optimálneho riadenia sa vyžaduje dobrá znalosť modelu, u ktorého je nevyhnutná identifikácia. Čím presnejšie máme popísaný model, tým lepšie môžeme navrhovať regulátor. Pri modelovaní systémov s viacerými vstupmi a výstupmi pomocou klasického prístupu Black-box nastáva problém. Nevieť totiž ako vyzerá, správa sa systém vo vnútri. Pozorujeme len vstupy a výstupy. Niekedy, je ale Black-box nevyhnutný. Vtip je v tom, že zvyčajne, ale niečo vieme viac a môžeme to zahrnúť do procesu identifikácie. Pri našej práci sme si preto zvolili prístup, ktorý sa nám zdá výhodnejší, Grey-box. Je to systém so zadanou vnútornou štruktúrou.

Práca guľička na rovine nám predstavuje systém podobný systému družici už spomínanej. Preto je veľmi zaujímavá a dáva nám možnosť lepšie pochopiť a niečo nové sa naučiť o riadení nestabilných systémov.

# Kapitola 2

## Identifikácia

Identifikácia je spracovanie meraní získaných na procese s cieľom zvoliť vhodný typ modelu, nájsť štruktúru a nakoniec identifikovať jeho parametry z nameraných dát. Identifikáciu zďaleka neuskutočňujeme len pre účely simulácie a návrhu riadiacich systémov, to je najmenšia časť aplikácií. Identifikácia sa uskutočňuje pre účely diagnostiky v biológii a lekárstve, meteorológii a iných oblastiach, kde formálne modely s parametrami získanými z dát umožňujú hlbšie poznanie systémov, predikciu chovania systému do budúcnosti či zmenu konštrukcie, pokiaľ ide o inžinierský systém [7]. Lineárne systémy sú veľmi často používané z hľadiska jednoduchého a pohodlného riadenia oproti nelineárnym systémom. Preto sa snažíme nelineárne systémy linearizovať v okolí nejakého pracovného bodu, aby sme ich následne mohli identifikovať a riadiť ako lineárne.

Poznáme niekoľko druhov identifikácií, spomenieme si niekoľko základných z nich. Napríklad by sme mohli odvodiť teoretický popis z fyzikálnej podstaty a následne uskutočniť nejaké experimenty na systéme, tak aby sme získali všetky potrebné konštanty.

Ďalej by to mohlo byť meranie frekvenčnej charakteristiky. Vieme, že frekvenčná charakteristika je v podstate prenos, ktorý plne popisuje lineárny systém.

Jedna z najslabších metód je identifikácia z prechodovej charakteristiky. Zmeriame teda prechodovú charakteristiku nejakého systému a následne hľadáme taký prenos, ktorý sa s nameranou charakteristikou zhoduje najviac [14].

V našej práci sa budeme zaoberať identifikáciou systému metódou Grey-box.

### 2.1 Model

Štúdium dynamiky reálnych systémov je tradične založené na modele systému. Modely systému sú zjednodušené, abstraktné nástroje používané pre predikciu chovania modelovaných systémov. Vďaka nim vieme dostatočne presne určiť ako sa systém bude chovať. Predikátor výstupu nám hovorí, ako sa bude chovať výstup pokiaľ poznáme históriu a budúci vstup. Zvyčajne je popísaný diferenčnými, či diferenciálnymi rovnicami.

Zložitý a veľmi detailný model spravidla obsahuje parametry, ktoré nie je možné číselne odhadnúť. Takéto modely bude veľmi ťažké analyzovať. Prehnanou snahou o

čo najvernejšie modelovanie detailného chovania môžu pominúť dôležité globálne charakteristiky. Na druhej strane veľmi zjednodušený model nebude schopný reprodukovat chovanie skutočného systému. Úlohou návrhára je preto navrhnúť čo najjednoduchší model, ktorý bude dostatočne presný na to, aby dokázal reprezentovať chovanie skutočného systému [7].

## 2.2 Analytické modelovanie(white box)

Matematický model procesu sa vytvára na základe fyzikálnych, chemických, matematických a iných zákonov často s ich zjednodušením. Zjednodušené modely sa potom skladajú do zložitých systémov, aby sa vytvoril čo najpresnejší model [12].

Spočiatku vychádzame z fyzikálneho modelu, kde máme systém popísaný pomocou fyzikálnych zákonov, javov. Následne rovnice upravujeme a dostávame diferenciálny popis modelu.

Nevýhoda spočíva v tom, že nikdy sa nám presne nepodarí popísať všetky podstatné fyzikálne javy. Nikdy sa nám nepodarí zmerať všetky potrebné konštanty.

### 2.2.1 Príklad

Azda jedným z najznámejších matematických modelov je matematické kyvadlo [1]. U matematického kyvadla sa skúma len hmotný bod zavesený na tenkom vlákne zanedbateľnej hmotnosti, zanedbáva sa odpor vzduchu pri pohybe kyvadla a trenia v závese, gravitačné pole sa považuje za homogénne. Matematické kyvadlo je mechanický oscilátor, ktorý pri dodaní počiatkovej energie voľne kmitá bez vonkajšieho pôsobenia. Pre malé výchylky platí, že toto kmitanie je harmonické, teda ho môžeme popísať pomocou funkcie sínus.

Matematický popis:

Výsledná sila pôsobiaca na hmotný bod je

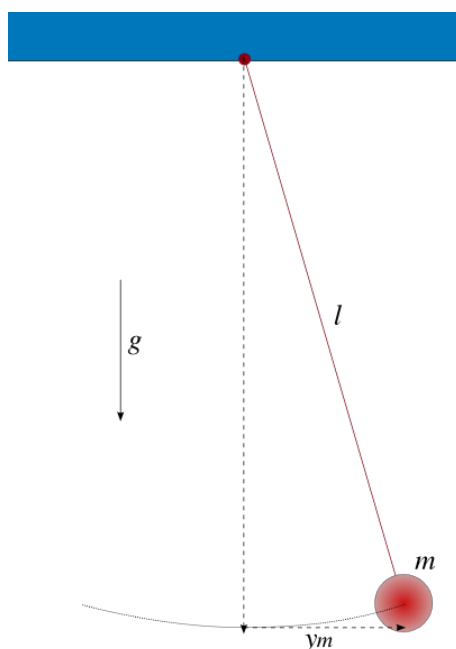
$$\mathbf{F} = m\mathbf{g} \sin \varphi$$

a diferenciálna rovnica pre popis pohybu kyvadla je z 2.Newtonova pohybového zákona

$$\ddot{\varphi} = -\frac{\mathbf{g}}{l} \sin \varphi$$

pokiaľ je maximálna výchylka z rovnovážnej polohy  $\varphi_{max}$  malá ( $< 5$  stupňov), ide funkciu sínus nahradit' lineárnou funkciou





Obrázek 2.1: Matematické kyvadlo

$$\sin \varphi \approx \varphi$$

Diferenciálna rovnica má preto jednoduchší tvar

$$\ddot{\varphi} = -\frac{g}{l}\varphi$$

## 2.3 Black-box

Pokiaľ nevieme nič o vnútornej štruktúre systému a nepoznáme fyzikálnu podstatu neostáva nám nič iné ako použiť black box model.

Čierna skrinka, takto by bol označený pojem v Slovenčine známy ako black box. Všeobecne sa jedná o označenie zariadenia, alebo obecné akéhokoľvek javu, u ktorého je zrejmé ako sa chová či prejavuje navonok, ale princípy, na ktorých funguje zostávajú pre nás utajené.

V typickom prípade má čierna skrinka vstupné a výstupné svorky. Experimentátor má možnosť privádzať na vstupné svorky rôzne napätia a signály a pritom merať signál na výstupných svorkách. Zo vzťahu medzi výstupným a vstupným signálom sa potom v priebehu funkcionálnej analýzy snaží odvodiť takzvaný funkcionálny model neznámeho systému. Z funkcionálneho modelu potom ide usudzovať na skrytú vnútornú štruktúru, tj. vytvárať hypotézy o možných štruktúrnych modeloch aj keď s konečnou istotou to nikdy nejde tvrdiť [5].

Uvedené postupy sú základom modelovania a funkcionálnej identifikácie systému, ktorá nachádza svoje uplatnenie ako v technických, priemyslových tak aj v iných obo-  
roch.

Schéma black boxu je znázornená na obrázku 2.2.



Obrázek 2.2: Black box

## 2.4 Grey-box

### 2.4.1 Úvod

Matematický model systému je základným stavebným prvkom pre väčšinu riadiacich algoritmov. S príchodom moderných metód riadenia, založených na optimalizácii vstupných a výstupných trajektórií, je dobrý model nezbytnou súčasťou kvalitného návrhu regulátoru.

Modely zaobstarané len na základe vstupno-výstupných dát pri neurčitej vnútornej štruktúre, tzv. black-box [8], môžu byť ľahko vzdialené fyzickej realite a častokrát dobre popisujú systém len na krátkom horizonte predikcie. Tento fakt vyplýva z podstaty používaných *prediction error method* (PEM) identifikácie. Zvyčajne máme k dispozícii viac informácií o zadanej sústave (statické zesílenie v danej vetvy, kmitavá/nekmitavá odozva), ktoré sa dajú s výhodou použiť pre zúženie súboru hľadaných modelov. Popis systému pomocou *grey-box* umožňuje ľahko zahrnúť tieto aprioritné informácie priamo do procesu identifikácie, čím sa dá získať fyzikálne opodstatnený model, ktorý bude lepšie predikovať v otvorenej smyčke.

K modelovaniu zvyčajne používame experiment, kedy vybudíme odozvu systému na zadaný vstupný signál. Úlohou identifikácie je potom nájsť taký popis systému, aby chyba medzi skutočným výstupom a výstupom vygenerovaným modelom bola čo najmenšia.

### 2.4.2 Teoretický základ

V nasledujúcom budeme uvažovať lineárny, časovo invariantný stochastický systém s diskretným meraním vstupu, popísaný v inovačnom tvare [11]

$$\mathbf{x}(k+1) = \mathbf{A}\mathbf{x}(k) + \mathbf{B}\mathbf{u}(k) + \mathbf{L}\mathbf{e}(k) \quad (2.1)$$

$$\mathbf{y}(k) = \mathbf{C}\mathbf{x}(k) + \mathbf{D}\mathbf{u}(k) \quad (2.2)$$

kde  $k \in \mathbb{R}$  je diskretný čas  $T_s$  ( $t = kT_s$ ),  $\mathbf{x} \in \mathbb{R}^{n_x}$  stav systému,  $\mathbf{y} \in \mathbb{R}^{n_y}$  meraný výstup,  $\mathbf{u} \in \mathbb{R}^{n_u}$  deterministický vstup a  $\mathbf{e} \in \mathbb{R}^{n_y}$  biely, gaussovský šum s nulovou strednou hodnotou a kovariančnou maticou

$$E\{\mathbf{e}(k)\mathbf{e}(k)^T\} = \mathbf{R}. \quad (2.3)$$

Takto popísaný systém umožňuje nezávislo modelovať deterministickú a stochastickú časť. Identifikácia systému vyššie popísaného ako *black – box* modelu je pomerne náročnou úlohou.

Tu predpokladáme, že matice

$$\mathbf{A} = \mathbf{A}(\theta), \mathbf{B} = \mathbf{B}(\theta), \mathbf{C} = \mathbf{C}(\theta), \mathbf{D} = \mathbf{D}(\theta), \mathbf{L} = \mathbf{L}(\theta) \quad (2.4)$$

majú pevne zadanú štruktúru a sú parametrizované vektorom parametrov  $\theta$ . Štruktúra matíc vychádza z fyzikálneho popisu systému a definuje hľadaný *grey – box*.

Naším cieľom je pre takto zvolenú štruktúru nájsť optimálnu hodnotu parametrov  $\theta$  takú, aby sme dosiahli čo najlepšie jednokrokovú predikcie výstupu.

## 2.5 Najmenšie štvorce

Metóda najmenších štvorcov je matematicko štatistická metóda spracovania zložitých dát, pomocou ktorej sa dá hľadať zjednodušený model: Umožňuje nájsť vhodnú aproximačnú funkciu pre dané empiricky zistené hodnoty. Hľadaná funkcia musí byť kombináciou vopred známych funkcií, metóda umožní vypočítať ich koeficienty [2].

Výsledok sa dá následne použiť ako optimalizačné kritérium, napríklad ako cieľ regulácie. **Kritérium optimality** poskytuje mieru presnosti dát v porovnaní s danou hypotézou. Výberový proces je určený riešením, ktoré optimalizuje kritérium použité k ohodnoteniu alternatívnych hypotéz [3]. Metóda najmenších štvorcov slúži k nalezaniu takého riešenia, aby súčet druhých mocnín chýb nájdeného riešenia bol minimálny, teda aby súčet štvorcov odchýlok bol najmenší.

Metódu najmenších štvorcov taktiež je možné použiť pre model ARX.

Často sa k aproximácii riešenia používa lineárna funkcia. V tomto prípade sa s touto metódou stretávame pod názvom lineárna regresia [2].

V tejto časti podkapitoly je vysvetlená lineárna regresia pomocou najmenších štvorcov. Teraz uvažujme o nejakom modeli (napríklad statického systému) popísaný lineárnou rovnicou v tvare

$$a_1x_1 + a_2x_2 \dots + a_nx_n = y$$

$x_i$  - veličina, ktorej hodnoty presne poznáme

$a_i$  - neznáme parametre

$y$  - veličina, ktorej hodnoty meriame

Cieľom nášho snaženia je nájsť parametra  $a_i$  v takto popísanom systéme. Ako už bolo spomenuté, hodnoty veličín  $x_i$  a  $a_i$  sú známe. Predpokladáme, že hodnoty veličín  $x_i$  poznáme presne, ale hodnoty veličín  $y$  sú zaťažené chybou merania

$$y = \hat{y} + e$$

kde ďalej predpokladáme, že chyba merania  $e$  je náhodná veličina s normálnym rozložením

$$e \approx N(0, \sigma_e^2)$$

Pokiaľ uskutočníme viac pozorovaní na systéme, získame sústavu niekoľkých rovníc

$$\begin{aligned} a_1x_1^1 + a_2x_2^1 + \dots + a_nx_n^1 &= y^1 \\ a_1x_1^2 + a_2x_2^2 + \dots + a_nx_n^2 &= y^2 \\ &\vdots \\ a_1x_1^k + a_2x_2^k + \dots + a_nx_n^k &= y^k \end{aligned}$$

Získali sme sústavu  $k$  rovníc o  $n$  neznámych. Túto sústavu môžeme zapísať maticovo kde

$$A = \begin{bmatrix} x_1^1 & x_2^1 & \dots & x_n^1 \\ x_1^2 & x_2^2 & \dots & x_n^2 \\ \vdots & \vdots & \vdots & \vdots \\ x_1^k & x_2^k & \dots & x_n^k \end{bmatrix} \quad b = \begin{bmatrix} y^1 \\ y^2 \\ \vdots \\ y^k \end{bmatrix} \quad x = \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_k \end{bmatrix}$$

Cieľom je nájsť neznámeho vektoru  $\mathbf{x}$ . Ak však máme viac rovníc ako neznámych, teda  $k > n$ , sústava je preurčená a obecně neexistuje exaktné riešenie. Správne označenie takéjto sústavy rovníc teda je

$$\mathbf{Ax} \approx \mathbf{b}$$

Existuje niekoľko možných definícií "najlepšieho" riešenia. Uvedený spôsob riešenia vychádza z predpokladu, že chybou je zaťažená len jedna veličina  $y$  a teda aj vektor  $\mathbf{b}$ . Maticu dát  $\mathbf{A}$  poznáme presne. Definícia najmenších štvorcov v tomto zmysle je potom nasledujúca

$$\min_{x \in R^n} \|\mathbf{Ax} - \mathbf{b}\|_2$$

Každé  $\mathbf{x}$ , ktoré minimalizuje uvedené kritérium sa potom nazýva *linear least squares solution*. K nájdeniu riešenia môžeme pristúpiť viacerými spôsobmi. Nižšie popísanú máme jednu z nich.

### 2.5.1 Riešenie LS pomocou pseudoinverzie

Predpokladáme sústavu rovníc

$$\mathbf{Ax} \approx \mathbf{b}$$

Obe strany rovnice môžeme vynásobiť maticou  $\mathbf{A}^T$  a dostávame

$$\mathbf{A}^T \mathbf{Ax} \approx \mathbf{A}^T \mathbf{b}$$

Keď sa dobre pozrieme na maticu  $\mathbf{A}^T \mathbf{A}$  zistíme, že ich rozmer je  $n \times n$ . To znamená, že sústava vyššie uvedená má  $n$  rovníc o  $n$  neznámych. Z toho vyplýva, že sústava rovníc bude mať jednoznačné riešenie a znamienko  $\approx$  môžeme nahradiť rovnosťou. Po vyjadrení  $\mathbf{x}$  získame rovnaké riešenie ako pri priamej minimalizácii normy (za predpokladu, že  $\text{rank}(\mathbf{A}) = n$ )

$$\mathbf{x}_{LS} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b}$$

### 2.5.2 Príklad

Teraz si spolu pozrieme riešenie jednoduchého príkladu na použitie najmenších štvorcov. Uvažujeme model systému v tvare priamky:

$$a_1x + a_2 = y$$

Na systéme sme spravili štyri pozorovania

–	$x$	$y$
1	–1	0.8
2	0	2.2
3	1	3.6
4	1	3.1

Zostavíme potrebné matice podľa definície uvedenej v podkapitole 2.5

$$\mathbf{A} = \begin{bmatrix} -1 & 1 \\ 0 & 1 \\ 1 & 1 \\ 0 & 1 \end{bmatrix} \quad \mathbf{b} = \begin{bmatrix} 0.8 \\ 2.2 \\ 3.6 \\ 3.1 \end{bmatrix} \quad \mathbf{x} = [ a_1 \ a_2 ]$$

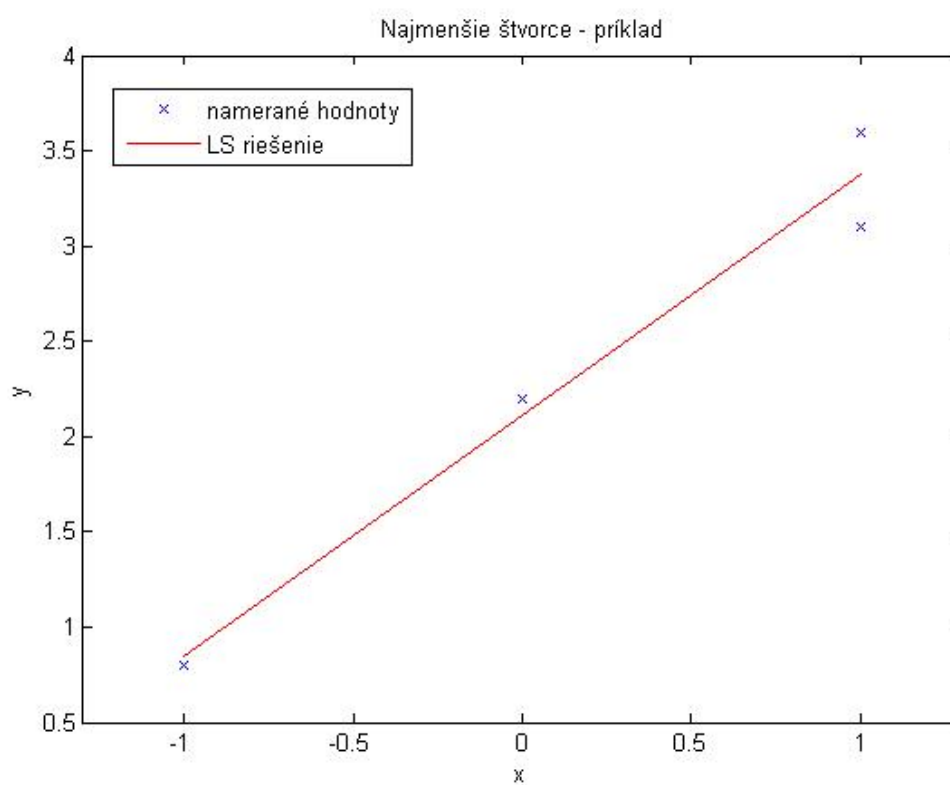
Výsledné riešenie statického systému je dané vzťahom

$$\mathbf{x}_{LS} = (\mathbf{A}^T \mathbf{A})^{-1} \mathbf{A}^T \mathbf{b} = \begin{bmatrix} 1.2636 \\ 2.1091 \end{bmatrix}$$

Výsledný model statického systému je daný rovnicou (viď obrázok)

$$1.2636x + 2.1091 = y$$

Takto bola popísaná metóda lineárnych najmenších štvorcov. Naša úloha, ale vedie na nelineárne najmenšie štvorce. Riešenie spočíva v iteratívnom riešení pomocou optimalizačného solvra. Nie je zaistená konvergencia k globálnemu optimu. Dôležitá je správna inicializácia počiatočných podmienok (počiatočný odhad). Čo nie je v našom prípade až také obmedzenie, keď vezmeme v úvahu, že máme nejakú predstavu o hodnote parametrov z fyzikálneho modelovania.



Obrázek 2.3: Najmenšie štvorce - príklad





# Kapitola 3

## Grey-Box

Identifikácia pomocou metódy Grey-box je veľmi dobre použiteľná pri identifikácii systémov s čiastočne známou vnútornou štruktúrou. Zvyčajne máme niekoľko známych údajov o danej sústave, ktoré sme získali pomocou meraní. Tieto údaje môžeme použiť pre zúženie súboru hľadaných modelov. Popis systému pomocou Grey-box umožňuje ľahko zahrnúť tieto informácie do procesu identifikácie.

Naším cieľom je pre zvolenú štruktúru (2.1)(2.2) nájsť optimálnu hodnotu parametrov  $\theta$  takú, aby sme dosiahli čo najlepšiu viackrokovú predikciu výstupu. Za predpokladu normálnych bielych šumov sa jedná o maximálne vierohodný odhad (ML)[10]. V našom prípade formulujeme tvar kritéria lineárnych systémov pre ML

$$J = \sum_{t=1}^N (e_P(t))^2 \quad (3.1)$$

$$\mathbf{e}_P(t) = \mathbf{y}(t) - \mathbf{C}\mathbf{x}_P(t) + \mathbf{D}\mathbf{u}(t) \quad (3.2)$$

$$\mathbf{x}_P(t+1) = \mathbf{A}\mathbf{x}_P(t) + \mathbf{B}\mathbf{u}(t) + \mathbf{L}\mathbf{e}_P(t) \quad (3.3)$$

kde  $e_P(t)$  je chyba predikcie a  $\mathbf{x}_P(t)$  je stav systému.

Optimálna hodnota  $\theta^*$  teda podľa ML odhadu bude

$$\theta^* = \arg \min_{\theta} J \quad (3.4)$$

### 3.1 Optimalizácia

Naša práca vedie na úlohu nelineárnych najmenších štvorcov. Optimalizačný problém je potrebné riešiť numerickými metódami, ktoré iteratívne vyhodnocujú stratovú funkciu (3.1) a aktualizujú hodnotu  $\theta$ . Pri inicializácii algoritmu je preto nevyhnutné udať počiatočný odhad parametrov  $\theta_0$ . Kritérium pravdepodobne nebude konvexné a teda nemáme istotu, že nájdené minimum bude globálny minimum. Odhad je závislý na správnej voľbe

počiatočných podmienok. Keďže pri modelovaní pomocou grey-box vychádzame z fyzikálneho popisu systému, ide tieto podmienky stanoviť relatívne správne. Pre vlastnú numerickú optimalizáciu používame algoritmus NLSQ Levenberg-Marquardt [13], s obmedzením na hodnotu parametrov v tvare

$$\theta_{min} \leq \theta \leq \theta_{max}$$

### 3.1.1 Optimalizačný solver

V tejto časti sa budeme zaoberať lsqnonlin solverom použitým v Matlabe.

Rieši problém nelineárnych najmenších štvorcov, zahŕňajúc problémy nelineárnej data-montáži. Radšej ako počítanie hodnôt  $\sum_{i=1}^n \|f_i(x)\|_2^2$  (suma štvorcov), lsqnonlin požaduje užívateľom definovanú funkciu k spočítaniu vektoru-ocenujúcej funkcie

$$f(x) = \begin{bmatrix} f_1(x) \\ f_2(x) \\ \vdots \\ f_n(x) \end{bmatrix}$$

Pričom funkcia  $f(x)$  počíta chybu predikcie  $e_P(t)$ (3.2). Potom, vo vektorovej oblasti, môžeme novo formulovať problém optimalizácie ako

$$\min_x \|f(x)\|_2^2 = \min_x (f_1(x)^2 + f_2(x)^2 + \dots + f_n(x)^2) \quad (3.5)$$

kde  $\mathbf{x}$  je vektor a  $f(x)$  je funkcia, ktorá vracia vektor hodnôt.

$x = lsqnonlin(fun, x0, lb, ub, options)$  minimalizuje s optimalizačnými možnosťami špecifikovanými v štruktúre *options*. Pužijeme *optimset* k nastaveniu týchto možností. Prechod prázdnych matíc pre *lb* a *ub* ak neexistuje hranica.

$x = lsqnonlin(problem)$  najde minimum pre *problem*, kde problém je štruktúra popísaná v stupných argumentoch.

### 3.1.2 Vstupné argumenty

Funkcia argumenty zahŕňa obecný popis argumentov vstupujúcich do lsqnonlin. Táto časť poskytuje detaily o špecifikácii funkcie pre *fun*, *options* a *problem*:

**fun**

Funkcia, ktorej sumy štvorcov su minimalizované. *fun* je funkcia, ktorá akceptuje vektor  $x$  a vráti vektor  $F$ , objektívne vratenie funkcie v  $x$ . Funkcia *fun* môže byť špecifikovaná ako funkcia zapísaná v M-file.

$$x = lsqnonlin(@myfun, x0)$$

kde *myfun* je MATLAB funkcia ako

$$\text{function } F = \text{myfun}(x)$$

$F = \dots$  Počíta funkčné hodnoty v  $x$ .

**options**

Options poskytujú detaily o špecifikácii funkcie pre hodnoty *options*

**3.1.3 Možnosti**

Optimalizačné možnosti. Môžeme nastaviť alebo zmeniť hodnoty týchto možností použitím *optimset* funkcie. Niektoré možnosti platia pre všetky algoritmy, niektoré sú relevantné len keď používame large-scale algoritmus a ostatné sú relevantné len keď používame medium-scale algoritmus.

**3.2 Popis funkcie prederrSISO**

Funkciu *prederrSISO* sme definovali v matlabe a popisuje nám ako funguje výpočet chyby predikcie  $e_P(t)$  (3.2). *PrederrSISO* nám vlastne vyjadruje funkciu  $f(x)$  spomínanú v podkapitole optimalizačn solver. Na jej vytvorenie sme postupovali podľa týchto bodov:

- na základe Grey-box modelu a vstupných parametrov vektoru  $\theta$ (3.6) zostrojíme spojitý stavový model.
- diskretizácia spojitého popisu.
- pomocou známych výstupných a vstupných dát napočítame chyby predikcie  $e_P(t)$  (3.2)(3.2).

### 3.2.1 Volanie funkcie prederrSISO

V prvom kroku sme načítali dáta, ktoré musia byť spojité. Inicializovali sme počiatkové parametre na hodnoty,

$$\begin{aligned} T &= 0,18 \\ K_b &= 3/5 * g \\ K_m &= 0,202 \\ T_s &= 0,15 \end{aligned}$$

ktoré sme si predtým zmerali, odvodili a nastavili. Ďalej voláme funkciu prederrSISO, ktorá nám dokáže spočítať vektor chýb predikcie  $\mathbf{EP}$ . Funkcia pracuje následovne. Vstupným argumentom funkcie prederrSISO je parameter  $\theta$ (3.6), ktorým parametrizujeme rovnice stavového popisu (3.9-3.12).

$$\theta = \begin{bmatrix} T \\ K_b \\ K_m \\ \mathbf{L} \\ \mathbf{x} \end{bmatrix} \quad \mathbf{L} = \begin{bmatrix} L1 \\ L2 \\ L3 \end{bmatrix} \quad \mathbf{x} = \begin{bmatrix} x1 \\ x2 \\ x3 \end{bmatrix} \quad (3.6 - 3.8)$$

kde  $T$  je časová konštanta,  $K_b$  a  $K_m$  su zosílenia,  $\mathbf{L}$  je injekčná matica,  $\mathbf{x}$  je stavový vektor. Pre počiatkové podmienky sú hodnoty matice  $\mathbf{L}$  a vektoru  $\mathbf{x}$  nulové. Systém ďalej diskretizujeme a následne vypočítame chybu predikcie podľa vzťahu (2.1)(2.2). Výstupným parametrom funkcie je vektor chýb predikcie  $\mathbf{EP}$ . Po spočítaní chýb predikcie spustíme lsqnonlin optimalizačný solver.

### 3.2.2 Validace modelu

Aby sme vedeli nakoľko máme presný model pozorujeme skutočný, simulovaný a predikovaný priebeh. Predikovaný priebeh sme dostali tým, že model má v spätnej väzbe injekčnú maticu  $\mathbf{L}$  (3.6). Simulovaný je otvorená slučka, kde do modelu vstupuje len samotný vstup.

#### Funkcia fit

Fit funkcia nám určuje mieru zhody daných 2 kriviek. Určujeme ju v %. V našom prípade sa budeme pozerať na mieru zhody skutočného (reálneho) a simulovaného priebehu. Funkcia je definovaná:

$$Fit = 100 \left( 1 - \frac{\|Y - Y_s\|_2^2}{\|Y\|_2^2} \right) \quad [\%] \quad (3.13)$$

kde  $Y$  je skutočný priebeh a  $Y_s$  je simulovaný [11].

### 3.3 Príklad

V nasledujúcom príklade si názorne demonštrujeme použitie metódy Grey-box na identifikáciu parametrov. Ako model použijeme guľičku na tyči. Teda jeden vstup a jeden výstup. Model máme popísaný stavovým popisom:

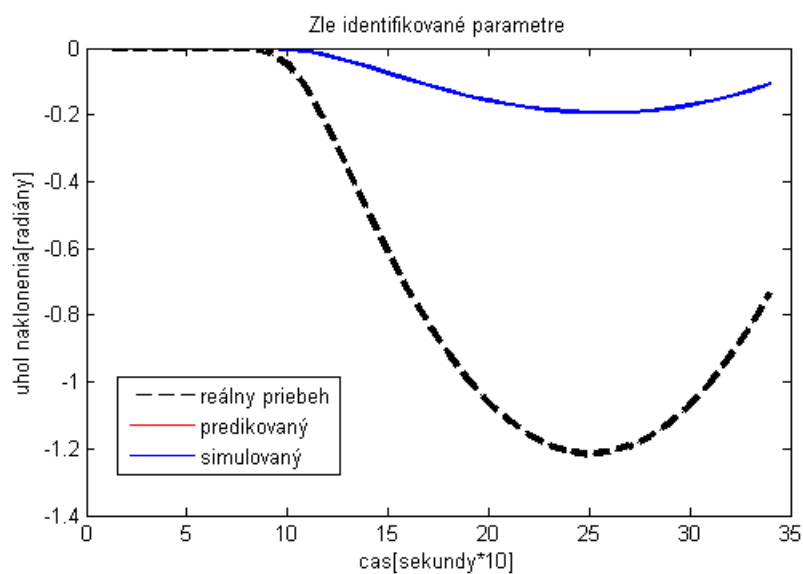
$$\mathbf{A} = \begin{bmatrix} -1/T & 0 & 0 \\ -Kb & 0 & 0 \\ 0 & 1 & 0 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} Km/T \\ 0 \\ 0 \end{bmatrix} \quad (3.9 - 3.12)$$

$$\mathbf{C} = [ 0 \quad 0 \quad 1 ] \quad \mathbf{D} = [ 0 ]$$

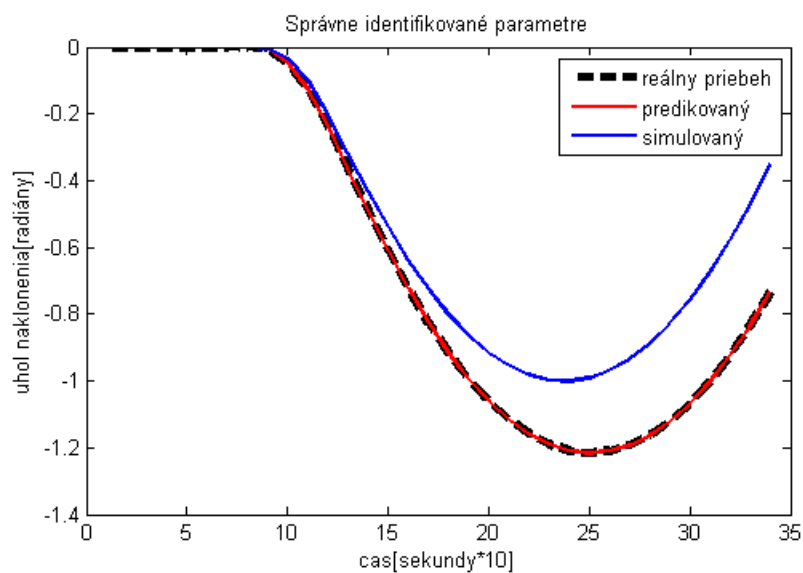
Keď už máme popísaný model a taktiež máme dáta, s ktorými môžeme pracovať môžeme identifikovať model. Na obrázku 3.1 sme naschvál zvolili, zle identifikované parametre, aby sme následne mohli poukázať na obrázku 3.2 ako dobre pracuje identifikácia Grey-box. Teda správne určenie parametrov.

počiatočné parametre	skutočné	správne, identifikované
$T = 0,5$	$T = 0,18$	$T = 0,1911$
$K_b = 5,259$	$K_b = 3/5g$	$K_b = 25,3784$

Vzorkovacia perióda  $T_s = 0,15s$  a parameter  $Km$  sme zafixovali na hodnotu  $Km = 0.202$ . Mohli sme si to dovoliť pretože vhl'adom k výstupu figuruje len jedno zosílenie  $K$ , pre ktoré platí  $K = K_m K_b$ , čo odpovedá  $u[-] \rightarrow$  náklon [radiány]. Pre počiatočné podmienky sú hodnoty matice  $\mathbf{L}$  a vektoru  $\mathbf{x}$  nulové. Na obrázku 3.1 sa prekrýva predikovaný a simulovaný priebeh, kvôli tomu, že matica  $\mathbf{L}$  je nulová. Na overenie zhody priebehov sme použili funkciu Fit (3.13). Pri nesprávne identifikovaných parametroch je rovná 28,20 % a pri správne identifikovaných je 94,6399 %. Po identifikácii zistujeme, že sa reálny a predikovaný priebeh výrazne zhodujú.



Obrázek 3.1: Zle identifikované parametre



Obrázek 3.2: Správne identifikované parametre

# Kapitola 4

## Popis systému guľička na rovine

### 4.1 Teoretický model guľička na rovine

Na odvodenie základného dynamického systému modelu vynímajúc servosystém pre naklonenie roviny, ktorý bude modelovaný osobitne, sa môže použiť variačná modelovacia metóda. Základná forma Euler-Lagrangeovej rovnice je

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

kde premenné majú nasledovný význam

$q_i$  je  $i$ -ty zovšeobecnený parameter

$\dot{q}_i$  je derivácia prvého rádu z  $i$ -tého zovšeobecneného parametra

$T$  je kinetická energia systému

$V$  je potenciálna energia systému

$Q_i$  je  $i$ -ta zovšeobecnená sila

Systém má štyri stupne voľnosti, dva v pohybe guľičky na rovine a dva stupne voľnosti v sklone roviny. Pre pozíciu guľičky sa ako všeobecné súradnice vybrali koordináty  $x$  a  $y$ . Pre sklon roviny boli vybraté uhly  $\alpha$  a  $\beta$ . Naklonenie roviny je ovládané zovšeobecnými krútiacimi momentmi  $\tau\alpha$  a  $\tau\beta$  pôsobiacimi na rovinu v príslušnom smere. A teda zjednodušené súradnice budú tieto premenné:

$$q_1 = x \quad q_2 = y \quad q_3 = \alpha \quad q_4 = \beta$$

Nasledujúce systémové premenné a parametre boli zadané:

$x, y$	súradnice pozície guľičky na rovine	[m]
$\mathbf{r}$	polohový vektor guľičky	[m]
$\mathbf{v}$	vektor rýchlosti guľičky	[m/s]
$r_b$	polomer guľičky	[m]
$\omega$	vektor uhlovej rýchlosti rotujúcej guľičky	[rad/s]
$\Omega$	vektor uhlovej rýchlosti rotujúcej roviny	[rad/s]
$\alpha, \beta$	uhly naklonenia roviny	[rad]
$I_b$	zotrvačnosť guľičky	[kg.m <sup>2</sup> ]
$I_p$	zotrvačnosť roviny	[kg.m <sup>2</sup> ]
$m$	hmotnosť guľičky	[kg]

Na zjednodušenie odvodenia dynamického systému sme uvažovali systém s týmito predpokladmi:

- kontakt guľičky s rovinou nie je prerušený za žiadnych okolností
- šmýkanie guľičky na rovine nie je dovolené
- všetky trecie sily a krútiace momenty sú zanedbané
- uhly roviny a rozsah obmedzenia nie sú zvažované

Celková kinetická energia systému je

$$T = T_p + T_b = \frac{1}{2}[(I_p + I_b)(\dot{\alpha}^2 + \dot{\beta}^2) + m(\dot{\beta}^2 y^2 + 2\dot{\alpha}\dot{\beta}xy + \dot{\alpha}^2 x^2) + (m + \frac{I_b}{r_b^2})(\dot{x}^2 + \dot{y}^2)]$$

kde  $T_p$  je kinetická energia roviny vrátane loptičky umiestnenej v polohe  $(x, y)$ , rotujúcej okolo svojho stredu.  $T_b$  je kinetická energia guľičky daná súčtom rotačnej energie vzhľadom na stred pohybujúcej sa guľičky a pohybovej energie guľičky [9].

## 4.2 Nelineárny model

Pri nelineárnom modeli budeme vychádzať z nasledujúcich rovníc

$$x : (m + \frac{5r_b^2}{r_b^2})\ddot{x} = m(\dot{\alpha}\dot{\beta}y + \dot{\alpha}^2 x) + m\mathbf{g} \sin \alpha \quad (4.1)$$

$$y : (m + \frac{5r_b^2}{r_b^2})\ddot{y} = m(\dot{\alpha}\dot{\beta}y + \dot{\alpha}^2 x) + m\mathbf{g} \sin \alpha \quad (4.2)$$

Tieto rovnice vyjadrujú matematicko-fyzikálny pohyb guľičky po naklonenej rovine. Vstupy do systému nie sú sily  $\mathbf{F}\alpha$  a  $\mathbf{F}\beta$ , ale priamo uhly  $\alpha$  a  $\beta$ . Je to kvôli tomu, že frekvencia krokovacieho motoru je nižšia ako akceleračný limit. Žiadne kroky motora



nemôžu byť stratené a veľkosť záťaže neovplyvňuje jeho pozíciu. Za tohoto predpokladu zanedbávame rovnice, ktoré nám popisujú ako je dynamika náklonu plošiny ovplyvnená vonkajšou riadiacou silou a pozíciou a rýchlosťou guľičky:

$$(I_p + I_b + mx^2)\ddot{\alpha} + m(\ddot{\beta}xy + \dot{\beta}\dot{x}y + \dot{\beta}x\dot{y} + 2\dot{\alpha}\dot{x}x) + m\mathbf{g}x \cos \alpha = \mathbf{F}_\alpha$$

$$(I_p + I_b + my^2)\ddot{\beta} + m(\ddot{\alpha}xy + \dot{\alpha}\dot{x}y + \dot{\alpha}x\dot{y} + 2\dot{\beta}\dot{x}x) + m\mathbf{g}x \cos \beta = \mathbf{F}_\beta$$

kde

$$(\dot{\alpha}\dot{\beta}y\dot{\alpha}^2x)m$$

odstredivý krútiaci moment vyplývajúci z rotácie roviny

$$(I_p + I_b + mx^2)\ddot{\alpha}$$

je krútiaci moment ako produkt zotrvačnosti z plošiny plus guľičky a uhlového zrýchlenia okolo jednej z osí.

$$(\ddot{\beta}xy + \dot{\beta}\dot{x}y + \dot{\beta}x\dot{y})m$$

je vplyv gyroskopických momentov.

$$2\dot{\alpha}\dot{x}x$$

je corioliosové zrýchlenie.

$$m\mathbf{g}x \cos \alpha$$

je moment ako produkt váhy guľičky.

Po úprave rovníc (4.1) a (4.2) dostávame

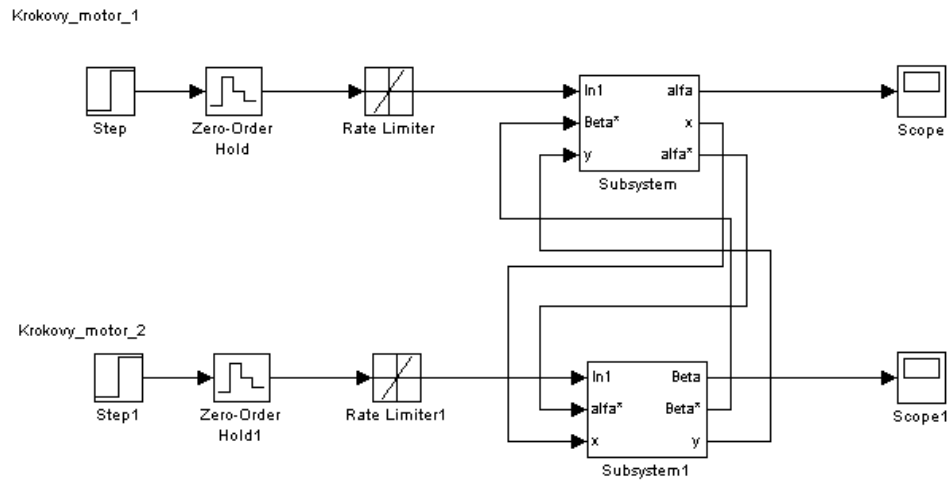
$$\ddot{x} = \frac{3}{5}\dot{\alpha}\dot{\beta}y + \frac{3}{5}\dot{\alpha}^2x - \frac{3}{5}\mathbf{g} \sin \alpha \quad (4.3)$$

$$\ddot{y} = \frac{3}{5}\dot{\alpha}\dot{\beta}y + \frac{3}{5}\dot{\beta}^2x - \frac{3}{5}\mathbf{g} \sin \beta \quad (4.4)$$

Na našom pracovnom modeli sme pracovali s ping-pongovou loptičkou. Preto sme na náš matematicko-fyzikálny nelineárny model aplikovali moment zotrvačnosti dutej guľičky. Teda

$$I_b = \frac{2}{3}mr_b^2$$

Na obrázku 4.1 vidíme simulinkovú schému nelineárneho modelu.



Obrázek 4.1: Nelineárny model

### 4.3 Linearizácia a stavový popis

Pre lineárne systémy používame stavové modely v tvare:

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}(t)\mathbf{x}(t) + \mathbf{B}(t)\mathbf{u}(t) & \mathbf{u}(t) \text{ sú vstupy} \\ \mathbf{y}(t) &= \mathbf{C}(t)\mathbf{x}(t) + \mathbf{D}(t)\mathbf{u}(t) & \mathbf{y}(t) \text{ výstupy} \\ & & \mathbf{x}(t) \text{ vnútorné stavy} \end{aligned}$$

kde  $\mathbf{A}(t)$  je matica systému  $\mathcal{R}(n \times n)$ ,  $\mathbf{B}(t)$  je matica riadenia rozmeru  $(n \times r)$ ,  $\mathbf{C}(t)$  a  $\mathbf{D}(t)$  sú výstupné matice rozmerov  $(m \times n)$  a  $(m \times r)$ . Rýdzo dynamický systém je taký, u ktorého je výstup  $y$  ovplyvnený vstupom  $u$  vždy s istým meškaním. To platí aj pre náš systém, teda matica  $\mathbf{D} = 0$ .

Pri linearizácii budeme vychádzať z nasledujúcich rovníc:

$$\dot{x}_2 = x_1$$

$$\dot{x}_1 = \ddot{x}_2 = \frac{3}{5}\dot{\alpha}\dot{\beta}y_2 + \frac{3}{5}\dot{\alpha}^2x_2 - \frac{3}{5}\mathbf{g} \sin \alpha$$

$$\dot{y}_2 = y_1$$

$$\dot{y}_1 = \ddot{y}_2 = \frac{3}{5}\dot{\alpha}\dot{\beta}y_2 + \frac{3}{5}\dot{\alpha}^2x_2 - \frac{3}{5}\mathbf{g} \sin \beta$$

$$\dot{\alpha} = -\frac{1}{T}\alpha - \frac{K_m}{T}u$$

$$\dot{\beta} = -\frac{1}{T}\beta - \frac{K_m}{T}u$$

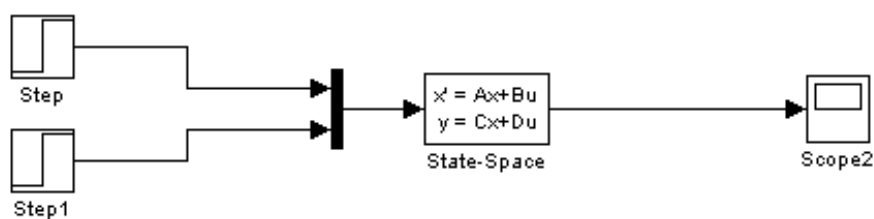
$$Y_x = \dot{X}_2$$

$$Y_y = \dot{y}_2$$

Linearizáciou nelineárneho modelu, skôr popísaného v okolí bodu  $x=0$ ,  $y=0$  a za predpokladu, že aj  $\dot{\alpha}=0$  a  $\dot{\beta}=0$  sme upravili nasledujúce rovnice. Dostali sme z nich následný stavový popis.

$$\mathbf{A} = \begin{bmatrix} -1/T & 0 & 0 & 0 & 0 & 0 \\ 0 & -1/T & 0 & 0 & 0 & 0 \\ -3/5g & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & -3/5g & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} K_m/T & 0 \\ 0 & K_m/T \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{bmatrix} \quad (4.5 - 4.8)$$

$$\mathbf{C} = \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \quad \mathbf{D} = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$



Obrázek 4.2: Lineárny model

## 4.4 Popis riadiacej karty humusoft MF 624

MF 624 reprezentuje viac-účelovú riadiacu kartu určenú pre potreby spojenia PC s externým zariadením, ktoré prijíma signály z reálneho sveta. MF 624 obsahuje 8 kanálov 14 bitových A/D prevodníkov so simultánnou sample/hold smyčkou, 8 nezávislých 14 bitových D/A prevodníkov, 8 bitový digitálny vstup a 8 bitový výstup, 4 kvadratické kódovacie vstupy s jedno-koncovým alebo s diferenciálnym rozhraním a 5 čítači. Karta je navrhnutá pre štandardné získavanie dát a riadenie aplikácií optimalizovaných pre použitie Real Time Toolbox v Simulinku. MF 624 plne podporuje 32 bitovú štruktúru pre rýchly prechodový výkon [9].

## 4.5 Servo systém

Náklon plošiny zabezpečujú 2 krokové servomotory. Tieto motory sú rovnaké, teda jeden krok odpovedá konštantnej zmene uhlu plošiny,  $\Delta\alpha = \Delta\beta$ . Krokový motor môžeme modelovať ako ideálny integrátor, ktorý integruje konštantnou rýchlosťou. Rýchlosť závisí na frekvencii pulzov podporovaných riadiacou kartou humusoft MF 624. Ovládače riadiacej karty krokových motorov a elektroniky karty samotnej vykazujú ďalšie známky nelinearity. V prvom prípade hodnota poslaná karte z prostredia Matlab je limitovaná na 1. Potom je hodnota zapísaná do zásobníku a obsah nemôže byť prepísaný pokiaľ karta generuje počet impulzov úmerných hodnote zaznamenananej v zásobníku. Toto mení tvar vstupného signálu ak sa vstupný signál servosystému mení rýchlejšie ako nominálna rýchlosť krokového motoru, potom motor nie je schopný nasledovať vstup a nastáva meškanie. Riešenie daného problému spočíva vo filtrácii vstupného signálu, tak aby servosystém bol schopný nasledovať filtrovaný signál. Najjednoduchší filter, odvodený priamo zo známej hodnoty rýchlosti krokovacieho motoru je rýchlostný obmedzovač ("rate limiter") [9].

Pre rýchlostný obmedzovač platí:

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

$$\begin{array}{ll} u_{\alpha f}(k) = +T_s R + u_{\alpha f}(k-1) & \text{pre rýchlosť} > +R \\ u_{\alpha f}(k) = -T_s R + u_{\alpha f}(k-1) & \text{pre rýchlosť} < -R \\ u_{\alpha f}(k) = u_{\alpha} & \text{inak} \end{array}$$

kde

$u_{\alpha}$  ... vstup do filtra

$u_{\alpha f}$  ... výstup z filtra

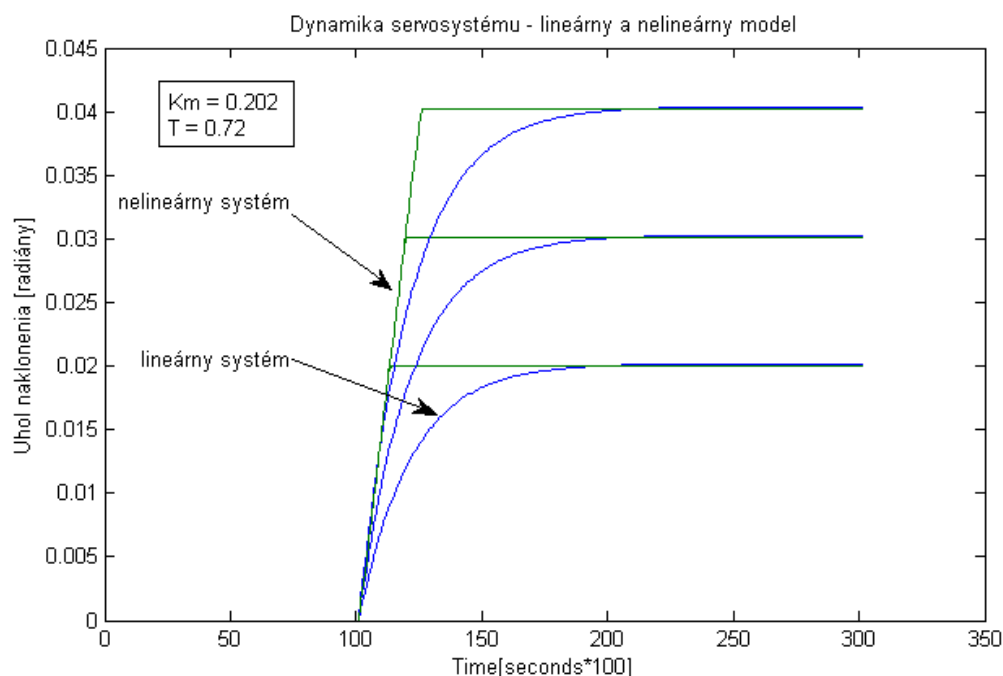
$k$  ... index času

$T_s$  vzorkovacia perioda

R ... prahová intenzita

Ak je prahová intenzita  $R$  rovná alebo nižšia nominálnej rýchlosti serva, potom je servo schopné kopírovať signál  $u_{\alpha f}$  a obsah riadiacej karty v zásobníku je pravidelne aktualizovaná( "update-ovaná").

Na obrázku 4.3 vidíme odozvu na skok pre rôzne veľkosti hodnôt. Je tam zobrazené správanie sa lineárneho a nelineárneho krokového motoru. Tvar krivky zodpovedá systému prvého rádu.



Obrázek 4.3: Odozva na skok

## 4.6 Identifikácia systému guľička na naklonenej rovine

V tejto časti kapitoly sa budeme zaoberať identifikáciou systému guľička na naklonenej rovine. Využívať budeme metódu grey-box spomínanú v kapitole Identifikácia a v kapitole Grey-box.

Rozhodli sme sa pre identifikáciu každého motora zvlášť, predpokladáme že sa nesprávajú úplne rovnako. Identifikovať každý motor zvlášť môžeme vďaka tomu, že sa nám systém rozdelil na dva nezávislé SISO systémy.

Počiatkové hodnoty parametrov sú:

$$T = 0,2037$$

$$K_b = 25,259$$

$$K_m = 0,202$$

$$T_s = 0,15$$

Matica  $\mathbf{L}$  (3.7) je nulová a vektor  $\mathbf{x}$  (3.8) je tiež. Teda guľička sa v počiatkových podmienkach nachádza v strede plošiny a plošina je vo vodorovnom stave.

### Identifikácia prvého systému

Teda motoru č.1 pôsobiaceho na náklon plošiny v jednej ose a na nej pôsobiacej guľičky. Na obrázku 4.4 vidíme vstupný a výstupný signál.

Na obrázku 4.5 vidíme, že predikovaný výstup kopíruje pomerne presne reálny výstup. No simulácia nie. Pribeh simulácie nesplýva v dôsledku pôsobiacich nelinearít. Čiastočne popisuje dynamiku systému. Z dôvodu zlej skupiny dát, sme teda nemohli správne identifikovať prvý motor.

### Identifikácia druhého systému

Teda motoru č.2 pôsobiaceho na náklon plošiny v druhej ose a na nej pôsobiacej guľičky. Na obrázku 4.6 vidíme vstupný a výstupný signál. Obrázok 4.6 nám ukazuje veľmi presnú identifikáciu. Simulácia aj predikcia sledujú výstup s veľmi veľkou zhodou až 99 % (3.13).

$$T = 0,2074$$

$$K_m = 0.202 \text{ fixovaný}$$

$$K_b = 25,2590$$

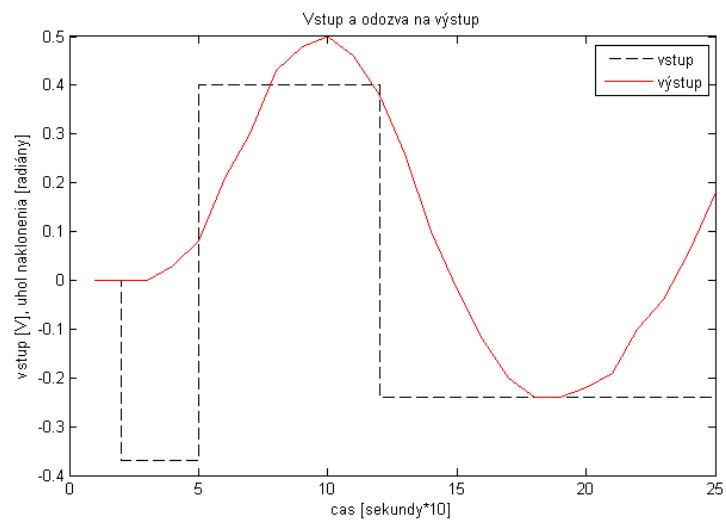
$$L_1 = 0,5208$$

$$L_2 = 3,9236$$

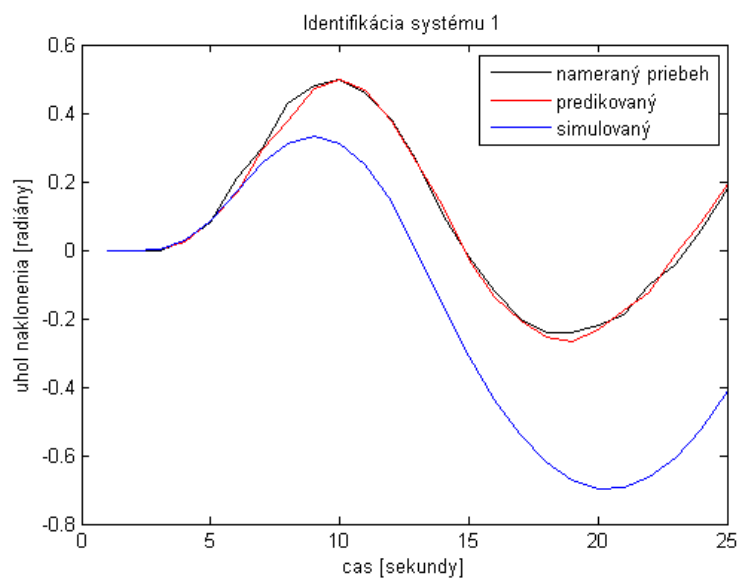
$$L_3 = 1,1089$$

### Záver

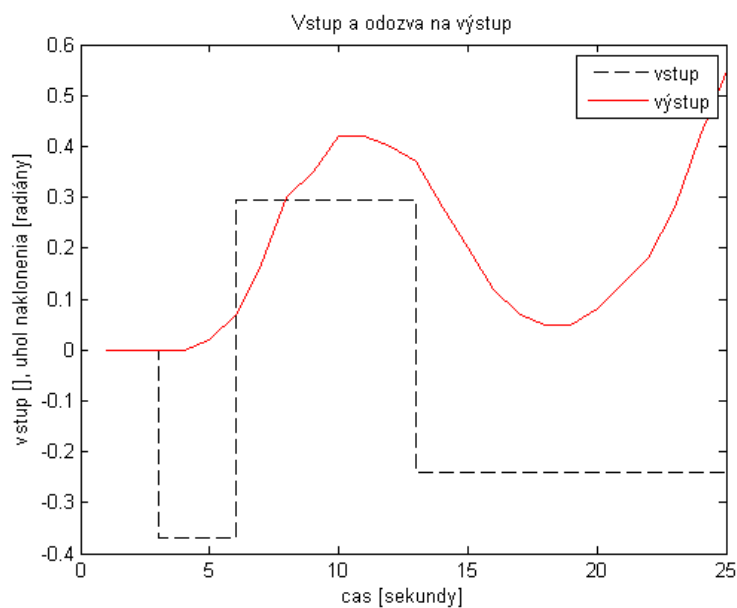
Pre nedostatok dát sme sa rozhodli urobiť model, ktorý bude popisovať oba motory súčasne. V skutočnosti by, ale bolo lepšie keby sme mohli identifikovať každý systém zvlášť ako sme mali v pláne.



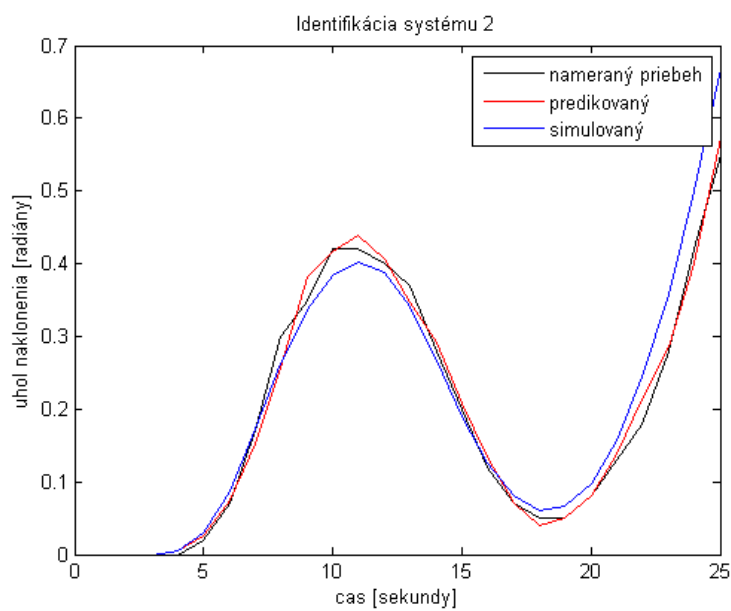
Obrázek 4.4: Vstup, výstup



Obrázek 4.5: Identifikácia systému 1



Obrázek 4.6: Vstup, výstup



Obrázek 4.7: Identifikácia systému 2



# Kapitola 5

## Návrh riadenia

Všetky doterajšie prípravy, teda identifikácia modelu viedli k jedinému, k riadeniu. Ak máme správne identifikovaný, model, môžeme prejsť k navrhovaniu funkčného regulátoru.

### 5.1 PID regulátor

K riadeniu väčšiny priemyslových procesov (a mnohých ďalších systémov) stačí PID regulátor, pokiaľ nie sú požiadavky na riadenie príliš vysoké. V našej práci sme použili PSD (Proportional-Summation-Differentiation) regulátor, teda diskretnú verziu PID regulátoru. Uplatňuje sa teda v diskretných systémoch. Pre diskretizáciu PID regulátorov sa najčastejšie používa metóda spätnej diferencie. Diskretná verzia spojitých verzií jednotlivých komponentov vyzerá nasledovne:

	Spojité verzia	Diskretná verzia
P	$P = K_p e(t)$	$P(t_k) = K_p e(t_k)$
I	$I(t) = \frac{K}{T_i} \int_{t_0}^t e(s) ds$	$I(t_k) = I(t_{k-1}) + \frac{KT_s}{T_i} e(t_k)$
D	$D(t) = -\frac{T_d}{N} \frac{dD}{dt} - KT_d \frac{dy}{dt}$	$D(t_k) = \frac{T_d}{T_d + NT_s} D(t_{k-1}) - \frac{KT_d N}{T_d + NT_s} (y(t_k) - y(t_{k-1}))$

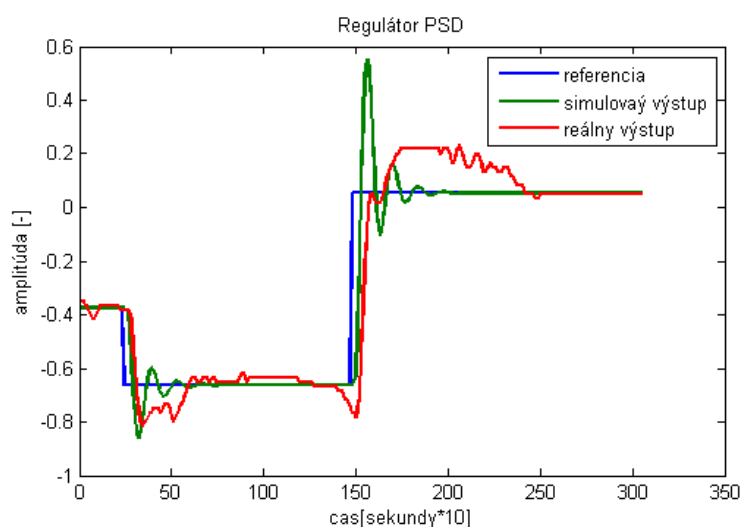
Konštanta N sa volí podľa potreby potlačenia zosilnenia derivačnej zložky na vysokých frekvenciách, typicky má hodnotu 10 až 30 [15].

Na obrázku B.2 (viď obrazová príloha) môžeme vidieť zapojenie nášho dvojdimenzionálneho PSD regulátora. Náš PSD regulátor bol navrhnutý empirickou metódou na skutočnom modeli. Postupne sme vyladili proporcionálnu, integračnú a derivačnú zložku až kým sme neboli spokojný s výsledkom. Hodnoty sú :

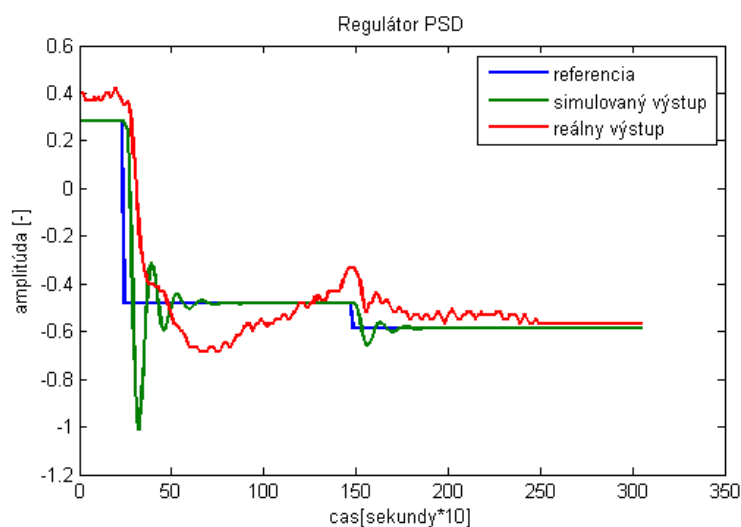
$$\begin{aligned} P &= 0,5 \\ I &= 0,09 \\ D &= 0,55 \end{aligned}$$

Schéma zapojenia regulátoru na danom systéme je na obrázku B.1 (viď obrazová príloha). Základné riadiace prvky sme mali od firmy Humusoft.

Náš regulátor nebol navrhnutý práve najideálnejšou metódou, empirickou. Preto po jeho navrhnutí sme chceli zistiť, ako sa bude správať na lineárnom modeli. Zistili sme, že prekmity sú dosť veľké. Pri správnejšom postupe (najprv simulovaní na lineárnom modeli a až potom na skutočnom systéme) by teda nami navrhnutý regulátor neprešiel. Na obrázkoch 5.1 - 5.2 vidíme, ako vyzeral simulovaný a reálny výstup na referenciu.



Obrázek 5.1: PSD regulátor zapojený na systém č.1



Obrázek 5.2: PSD regulátor zapojený na systém č.2

## 5.2 LQR (Lineárny kvadratický regulátor)

Lineárne stavové spätnoväzbové riadenie, ktoré je optimálne v niektorých prípadoch, môže byť určené ako riešenie zvané LQR problém. LQR problém bol skúmaný pre časovo-variantné a časovo-invariantné prípady. My sa zameriame na druhý prípad.

Uvažujme si časovo-invariantný lineárny systém:

$$\dot{\mathbf{x}} = \mathbf{F}\mathbf{x} + \mathbf{G}\mathbf{u}, \quad \mathbf{z} = \mathbf{H}\mathbf{x}$$

kde vektor  $\mathbf{z}^t$  reprezentuje premenné (signál, ktorý nás zaujíma), ktoré majú byť regulované, riadené k nule. Taktiež sa snažíme určiť  $u(t)$ ,  $t \geq 0$ , ktoré minimalizuje kvadratické kritérium

$$J(u) = \int_0^{\infty} [\mathbf{z}^T \mathbf{Q} \mathbf{z}(t) + \mathbf{u}^T \mathbf{R} \mathbf{u}(t)] dt$$

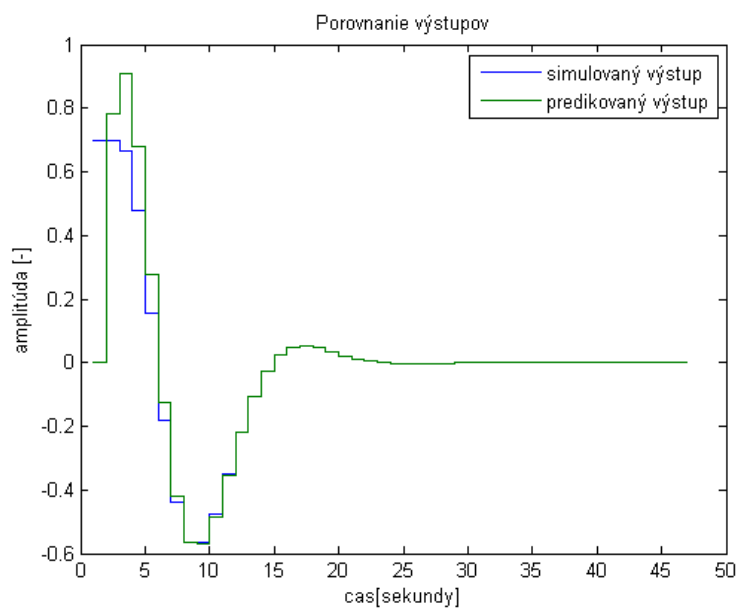
pre ľubovoľné počiatočné stavy  $x(0)$ . Váhové matice  $\mathbf{Q}$ ,  $\mathbf{R}$  sú reálne, symetrické a kladné,  $\mathbf{Q} = \mathbf{Q}^T$ ,  $\mathbf{R} = \mathbf{R}^T$ ,  $\mathbf{Q} > 0$ ,  $\mathbf{R} > 0$  [6]. Váhová matica  $\mathbf{Q}$  nám určuje do akej miery chceme mať presný výstup, a váhová matica  $\mathbf{R}$  nám určuje, aký veľký chceme mať akčný zásah. Optimálne riadenie má tvar  $u(t) = -\mathbf{K}_{opt}x(t) \rightarrow$  stavová spätná väzba, kde  $\mathbf{K}$  je stavové spätnoväzbové zosílenie.

Pri návrhu LQ regulátora budeme potrebovať pozorovateľa, ktorý vytvára odhad stavu sústavy miesto skutočného stavu sústavy, ktorý potom použijeme pre spätnú väzbu. Pozorovateľ má rovnaký vstup ako sústava a vedie doň aj výstup. Má rovnaký rád ako sústava. Póly pozorovateľa volíme 5x rýchlejšie, obvykle to sú 2 až 6-krát rýchlejšie [17]. Pre návrh pozorovateľa sme použili duálny Ackermanov vzorec. Pre výpočet stavového spätného zosílenia sme použili funkciu *dare* nadefinovanú v matlabe.

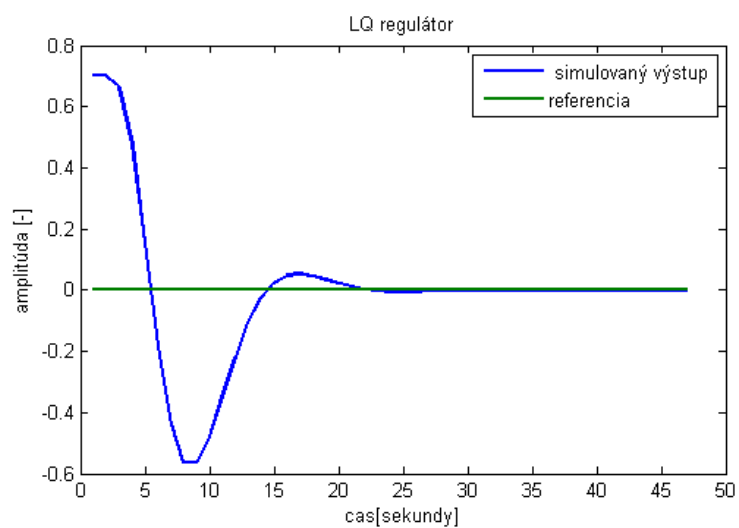
Na obrázku 5.3 vidíme simulovaný a pozorovateľom predikovaný výstup. Ako vidíme, predikovaný výstup pekne sleduje simulovaný. Na obrázku 5.4 vidíme, ako po zapojení LQ regulátora sleduje výstup referenciu. Zapojenie regulátora do lineárneho systému B.3 (viď obrazová príloha).

### Záver

Navrhnutý LQ regulátor na skutočnom systéme nefungoval. Na systém totiž pôsobili nelinearity. Strácali sa kroky motora a to viedlo k nesprávnemu odhadu nasledujúceho stavu a následne k nie najpresnejšej identifikácii. Správne riešenie by vyžadovalo rozšíriť systém o odhad poruchy, čo by však bolo nad rámec práce.



Obrázek 5.3: Výstupy



Obrázek 5.4: LQ regulátor

# Kapitola 6

## Záver

Bakalárska práca Identikace a návrh řízení pro model kulička na ploše si kládla za cieľ spracovať veľmi zaujímavú tému. Kvôli svojej náročnosti, ktorá presahuje rámec bežného bakalárskeho štúdia, si vyžadovala veľa času a trpezlivosti. Odmenou za vynaložené úsilie je však množstvo zaujímavých a dôležitých informácií, ktoré sa určite zídu v nasledujúcom štúdiu.

V práci sme sa venovali identikácii a návrhu riadenia pre systém guľička na rovine. Väčšiu váhu sme prikladali identifikácii. V identifikácii používame metódu Grey-box. Vybrali sme si ju kvôli možnosti ľahko zahrnúť aprioritné informácie (statické zosilnenie v danej vetve, kmitavá/nekmitavá odozva) priamo do procesu identikácie, čím je možné získať fyzikálne viac opodstatnený model, ktorý bude lepšie predikovať v otvorenej smyčke. Kvôli dôležitosti správneho modelu pre potreby neskoršej optimálnej regulácie sme venovali tejto časti práce väčšinu pozornosti a úsilia.

Pri identikácii sme rozdelili systém na dva SISO systémy. Neskôr sme ale zistili, že kvôli nedostatku dát musíme urobiť model, ktorý bude popisovať oba motory súčasne. Napriek skutočnosti, že by bolo lepšie identikovať každý systém zvlášť podľa pôvodného plánu, sme úspešne zvládli aj túto druhú možnosť. Výsledkom identifikácie je teda popísaný model, ktorý zohľadňuje všetky dôležité fyzikálne vlastnosti reálneho systému.

Druhou časťou nami vypracovanej práce bol návrh riadenia pomocou LQ alebo PID regulátora. PID regulátor sme navrhli empirickou metódou, ktorá sa ukázala ako najvhodnejšia. Úspešnosť použitej metódy dokazujú odozvy namerané na reálnom systéme. Navrhnutý LQ regulátor na skutočnom systéme nefungoval z dôvodu pôsobenia externých porúch. Strácali sa kroky motora a to viedlo k nesprávnemu odhadu nasledujúceho stavu a následne k zhoršeniu presnosti identikácie. Riešenie vzniknutých problémov vidíme v rozšírení o odhad poruchy. Keďže však táto problematika presahuje rámec našej práce, navrhujeme odhad poruchy ako vhodné a vítané rozšírenie realizovateľné v priebehu nasledujúcich nadväzujúcich prác.

Výsledkom práce Identikace a návrh řízení pro model kulička na ploše je teda ucelený model, ktorý svojim popisom s využitím pokročilých metód identifikácie veľmi verne vystihuje reálny systém. Predložený je spracovaný návrh dvoch regulátorov, ktoré boli otestované aj na reálnom modeli. Metódy spracovania a návrhu vychádzajú z moderných

trendov a je možné ich ďalej vylepšiť, rovnako ako aj modifikovať a použiť pri práci s inými modelmi v rámci výuky, výskumu či technickej praxe.

# Literatura

- [1] Matematické kyvadlo.  
[http://cs.wikipedia.org/wiki/Matematické\\_kyvadlo](http://cs.wikipedia.org/wiki/Matematické_kyvadlo).
- [2] Metoda nejmenších čtverců.  
[http://cs.wikipedia.org/wiki/Metoda\\_nejmenších\\_čtverců](http://cs.wikipedia.org/wiki/Metoda_nejmenších_čtverců).
- [3] Optimality criterion.  
[http://en.wikipedia.org/wiki/Optimality\\_criterion](http://en.wikipedia.org/wiki/Optimality_criterion).
- [4] Wattův odstředivý regulátor.  
[http://en.wikipedia.org/wiki/Wattův\\_odstředivý\\_regulátor](http://en.wikipedia.org/wiki/Wattův_odstředivý_regulátor).
- [5] Černá skříňka (kybernetika).  
[http://cs.wikipedia.org/wiki/Černá\\_skříňka\\_\(kybernetika\)](http://cs.wikipedia.org/wiki/Černá_skříňka_(kybernetika)).
- [6] M. Ansaklis. *Linear System Primer*.
- [7] Doc.Ing.P.Horáček. *Systémy a modely*, volume 1. Vydavatelství ČVUT, 2000.
- [8] G. G.E.P a Jenkins. *Time series analysis, forecasting and control*, volume 1. 1990.
- [9] Humusoft. Educational manual.
- [10] V. H. Jiří Řehoř. Identifikace grey-box modelu s praktickou aplikací v průmyslu a energetice.
- [11] L. Ljung. *System identification*, volume 1. Springer, 1987.
- [12] T. Math-Works. System Identification Toolbox – Getting Started Guide.
- [13] J. J. Mor. The Levenberg-Marquardt algorithm: implementation and theory, Lecture notes in mathematics. *Springer*, 630:105–116.
- [14] J. Roubal. Identifikace. page 1.
- [15] K. řídicí techniky ČVUT. PSD regulátor.  
[http://dce.felk.cvut.cz/sari/download/sri/cv11\\_psd\\_regulator.pdf](http://dce.felk.cvut.cz/sari/download/sri/cv11_psd_regulator.pdf).
- [16] M. Šebek. Systémy a řízení 1:Úvod. pages 7–9.
- [17] M. Šebek. Systémy a řízení 7:Observer.





# Dodatek A

## lsqnonlin solver

$x = \text{lsqnonlin}(fun, x0)$  začína v bode  $x0$  a hľadá minimum sumy štvorcov funkcie popísanej vo  $fun$ .  $fun$  by mal vracať vektor hodnôt a nie sumu štvorcov hodnôt. (Algoritmus implicitne sumuje a štvorcuje  $fun(x)$ .)

$x = \text{lsqnonlin}(fun, x0, lb, ub)$  definuje a nastavuje spodnú a hornú hranicu na vzorke premenných v  $x$ , takže toto riešenie je vždy v rozsahu

### problem

<i>objective</i>	Všeobecná funkcia
<i>x0</i>	Počiatkové podmienky
<i>lb</i>	Vektor spodnej hranice
<i>ub</i>	Vektor hornej hranice
<i>solver</i>	'lsqnonlin'
<i>options</i>	Možnosti štruktúry vytvorené s funkciou <i>optimset</i>

### A.0.1 Výstupné argumenty

Funkcia argumenty zahŕňa obecný popis argumentov vrátených `lsqnonlin`. Táto časť poskytuje detaily o špecifikácii funkcie pre *exitflag*, *lambda*, *output*:

#### exitflag

Integer identifikujúci ukončenie algoritmu	Dôvod ukončenia algoritmu <i>exitflag</i> .
1	Funkcia konvergujúca k riešeniu $x$ .
2	Zmena v $x$ bola menšia ako špecifikovaná tolerancia.
3	Zmena zvyšku bola menšia ako špecifikovaná tolerancia.
4	Veľkosť hľadaného smeru bola menšia ako špecifická tolerancia.
0	Počet iterácií prekročil <i>options.MaxIter</i> alebo počet hodnotení funkcie prekročil <i>options.FunEvals</i> .
-1	Algoritmus bol ukončený výstupnou funkciou.
-2	Problém je nerealizovateľný: hranice $lb$ a $ub$ sú nekonzistentné.
-4	Line vyhľadávanie nemohlo dostatočne znížiť $\ g\ $ pozdĺž súčasného vyhľadávacieho smeru.

**lambda**

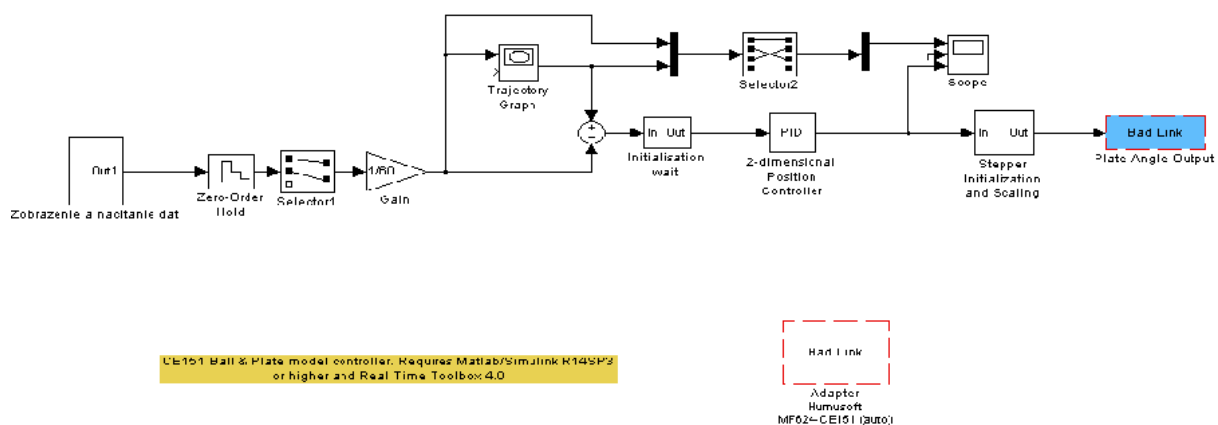
Štruktúra obsahujúca Lagrangeove	multiplikátory v riešení $x$ . Polia sú
<i>lower</i>	Spodná hranica $lb$
<i>upper</i>	Horná hranica $ub$

**output**

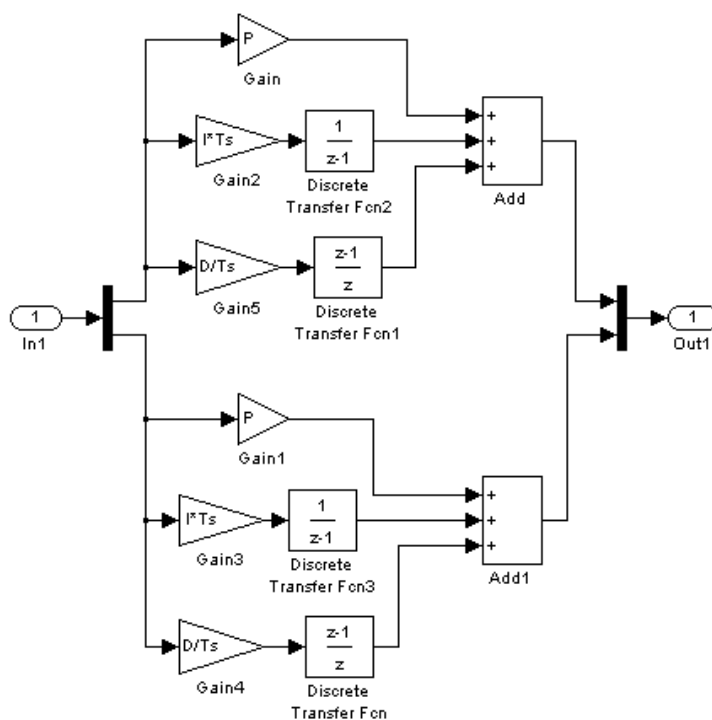
Štruktúra obsahujúca informácie o optimalizácii	Polia štruktúry sú
<i>firstorderopt</i>	Meranie optimality prvého rádu (large-scale algoritmus, [] pre ostatné)
<i>iterations</i>	Počet zobratých iterácií
<i>funcCount</i>	Počet hodnotení funkcie
<i>cgiterations</i> Celkový počet PCG iterácií	(large-scale algoritmus, [] pre ostatné)
<i>stepsize</i>	Finálne posunutie v $x$ (len medium-scale algoritmus)
<i>algorithm</i>	Použitý optimalizačný algoritmus
<i>message</i>	Výstupná správa

# Dodatek B

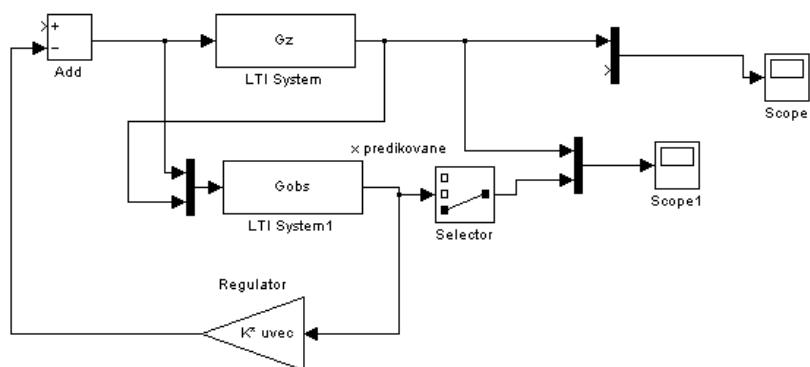
## Obrazová príloha



Obrázek B.1: Simulinková schéma zapojenia PSD do systému



Obrázek B.2: PSD regulátor



Obrázek B.3: Zapojenie LQ regulátora

# Dodatek C

## CD

- Súbory vytvorené v matlabe
- PDF bakalárskej práce Identikace a návrh řízení pro model kulička na ploše
- Naskenovanie oficiálneho zadania práce a podpísaného prehlásenia