

Západočeská univerzita v Plzni
Fakulta aplikovaných věd
Katedra kybernetiky

BAKALÁŘSKÁ PRÁCE

PLZEŇ, 2012

MATĚJ KOVÁŘÍK

PROHLÁŠENÍ

Předkládám tímto k posouzení a obhajobě bakalářskou práci zpracovanou na závěr studia na Fakultě aplikovaných věd Západočeské univerzity v Plzni.

Prohlašuji, že jsem bakalářskou práci vypracoval samostatně a výhradně s použitím odborné literatury a pramenů, jejichž úplný seznam je součástí.

V Plzni dne

.....

Poděkování

Tímto bych chtěl poděkovat svému vedoucímu bakalářské práce panu Prof. Ing. Miloši Schlegelovi CSc. Moje poděkování mu patří především za jeho trpělivost, za vedení práce ke správnému výsledku a čas věnovaný konzultacím, při kterých mi vždy pomohl daný problém řešit.

Anotace

Bakalářská práce se zabývá návrhem jednohého skákacího robotu se třemi rameny. Cílem bylo vytvořit model robota v prostředí MATLAB/SIMULINK/SimMechanics a dokázat robotu stabilizovat ve svislé poloze. Robot se dokáže stabilizovat pomocí řízení pohybu ramen.

This thesis deals with one-legged jumping robot with three arms. The aim was to create a model of the robot in MATLAB/Simulink/SimMechanics and be able to stabilize the robot in an upright position. The robot is able to stabilize by motion control of the arms.

Klíčová slova

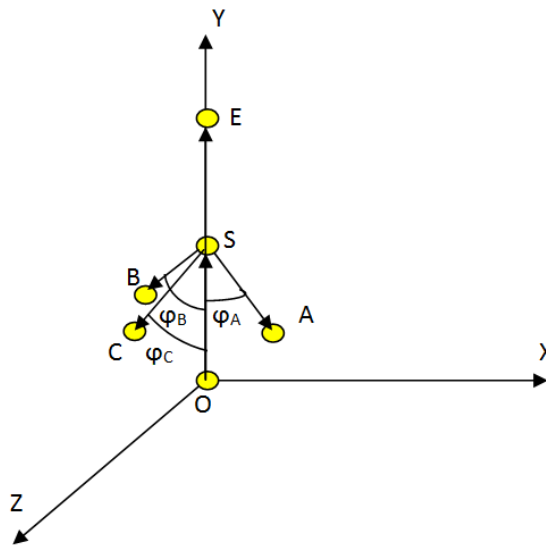
skákací robot, více-tělesový systém, modelování v SimMechanics, stabilizace robotu, stavový regulátor, návrh regulátoru, kritérium ISE

Obsah

1	Úvod	4
2	Metody vytváření matematických modelů více-tělesových systémů	6
2.1	Matice rotace	6
2.2	Těžiště soustavy hmotných bodů	7
3	Model skákacího robotu v SimMechanics	8
3.1	Funkční bloky MATLAB/SIMULINK/SimMechanics	8
3.2	Dvojité kyvadlo	9
3.3	Označení parametrů robotu	10
3.4	Určení vektorů systému	11
3.5	Skákací robot-varianta 1	12
3.6	Skákací robot - varianta 2	14
4	Analýza modelu	15
4.1	Vyjádření obecné polohy těžiště systému	15
4.2	Optimalizace parametrů podle speciálního případu	16
5	Návrh řídicího systému robotu	18
5.1	Návrh řízení dvojitého kyvadla	18
5.2	Návrh stavového regulátoru podle kritéria ISE	21
5.3	Stabilizace skákacího robotu v nakloněné poloze	22
6	Ověření funkčních vlastností	24
6.1	Simulace řízených kyvadel	24
6.2	Simulace stabilizace robotu	25
6.3	Libovolné naklonění těla robotu	27
7	Návrh funkčního vzorku	28
7.1	"Voice coil" actuátor	28
7.2	Piezo actuátor	30
8	Závěr:	31
8.1	Seznam zdrojů:	32

1 Úvod

V úvodu nastíníme problematiku skákacího robotu, kterou se budeme zabývat. Skákací robot se skládá z těla a tří ramen, která jsou připojena k tělu robota v polovině jeho délky. Ramena jsou k robotu připojena tak, že mezi sebou svírají úhel 120° . Hmotu těla a ramen budeme modelovat pomocí soustavy hmotných bodů. Model robotu by tedy měl vypadat takto:



Obrázek 1: Umístění bodů v soustavě souřadnic

Srovnáme-li tento model s modely, kterými se zabývají podobné články viz. [5] a [6], tento model skákacího robotu se v konstrukci modelu odlišuje. Modely v člancích viz. [5] a [6] uvažují robot, který je v počátečním stavu ve stabilní poloze. Tělo robotu stojí na plochem chodidle. Robot, kterým se zabývá tato práce je v počátečním stavu ve vratké poloze. Tělo robotu má bodový styk s podložkou. Proto musíme zajistit stabilizaci robota, pokud chceme, aby robot stál i při vychýlení z této polohy.

Nutnost stabilizovat robot můžeme řešit ve dvou základních situacích. První problém předpokládá, že tělo robota stojí na pevné ploše. V tomto případě chceme, aby se robot dokázal stabilizovat ve svislé poloze, ale taky chceme, aby se dokázal z této polohy vychýlit. Vychýlení ze svislé polohy lze dosáhnout řízením ramen. Pokud dokážeme robot stabilizovat v nakloněné poloze, můžeme pak s robotem skočit do dálky. Neboť ze svislé polohy se totiž dokáže robot odrazit pouze do výšky a dopadne ve stejném bodě, v kterém se odrazil. Na nechtěné vychýlení by měl robot reagovat tak, že se vrátí zpět do rovnovážné polohy. Nechtěné vychýlení je způsobeno vnější silou, například větrem.

Jako druhý problém uvažujeme situaci, kdy robot skočí do dálky. Při výskoku mělo tělo robotu určité naklonění vůči svislé ose. Pokud by skákací robot dopadl se stejným nakloněním, bylo by pravděpodobné, že po dopadu robot spadne v nakloněném směru. Proto bychom chtěli, aby se robot během letu natočil do takové polohy, pro kterou by se po dopadu na zem dokázal stabilizovat, například ve svislé poloze. Tento problém však v této práci řešen není.

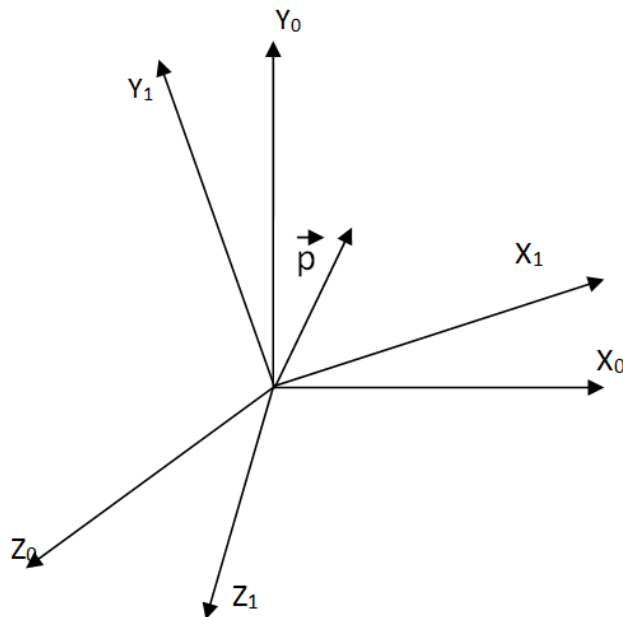
Vezmeme-li tedy problém stabilizace robotu na pevné ploše, navrhujeme řízení ramen takové, aby robot zůstal v rovnovážné poloze. Pro regulování ramen robotu použijeme stavový regulátor, který reguluje daný systém do "nuly". Pro návrh regulátoru lze použít metodu návrhu podle minimálního kritéria ISE nebo metodu návrhu podle umístění pólů systému.

Nezbytné pro vytvoření modelu v programovém prostředí SimMechanics je vytvořit popis robotu v soustavě souřadnic. Proto se tato práce věnuje v následující kapitole metodám vytváření matematických modelů. Abychom mohli navrhnout řízení k vytvořenému modelu v SimMechanicsu tak, aby splnilo určité požadavky na možné odchýlení od osy Y, je nutné provést optimalizaci parametrů robotu. Proto provedeme nejdříve analýzu modelu před návrhem řízení.

2 Metody vytváření matematických modelů více-tělesových systémů

2.1 Matice rotace

Definujeme 2 pravotočivé kartézské SS¹, které jsou určeny jednotkovými vektory \vec{x}_i , \vec{y}_i , \vec{z}_i . Obě soustavy souřadnic mají stejný počátek bod $[0,0,0]$.



Obrázek 2: Umístění bodů v soustavě souřadnic

Vektor \vec{p} má počáteční bod v počátku SS, proto lze vyjádřit jeho souřadnice v daných soustavách takto:

$$\vec{p} = a_0 \cdot \vec{x}_0 + b_0 \cdot \vec{y}_0 + c_0 \cdot \vec{z}_0 = a_1 \cdot \vec{x}_1 + b_1 \cdot \vec{y}_1 + c_1 \cdot \vec{z}_1 \quad (1)$$

Chceme určit vztah mezi souřadnicemi vektoru \vec{p} v soustavě S_{x_0, y_0, z_0} a v soustavě S_{x_1, y_1, z_1} . Rovnici (1) proto vynásobíme postupně všemi směrovými vektory soustavy S_{x_0, y_0, z_0} . Dostaneme tři rovnice:

$$\begin{aligned} a_0(\vec{x}_0, \vec{x}_0) + b_0(\vec{x}_0, \vec{y}_0) + c_0(\vec{x}_0, \vec{z}_0) &= a_1(\vec{x}_0, \vec{x}_1) + b_1(\vec{x}_0, \vec{y}_1) + c_1(\vec{x}_0, \vec{z}_1) \\ a_0(\vec{y}_0, \vec{x}_0) + b_0(\vec{y}_0, \vec{y}_0) + c_0(\vec{y}_0, \vec{z}_0) &= a_1(\vec{y}_0, \vec{x}_1) + b_1(\vec{y}_0, \vec{y}_1) + c_1(\vec{y}_0, \vec{z}_1) \\ a_0(\vec{z}_0, \vec{x}_0) + b_0(\vec{z}_0, \vec{y}_0) + c_0(\vec{z}_0, \vec{z}_0) &= a_1(\vec{z}_0, \vec{x}_1) + b_1(\vec{z}_0, \vec{y}_1) + c_1(\vec{z}_0, \vec{z}_1) \end{aligned} \quad (2)$$

¹Soustava souřadnic

Skalární součin rovnoběžných nebo totožných vektorů je 1. Vektory určující SS jsou ortonormální. Vyjedme-li z těchto skutečností, rovnice se zjednoduší a dají se zapsat v tomto maticovém tvaru:

$$\begin{bmatrix} a_0 \\ b_0 \\ c_0 \end{bmatrix} = \begin{bmatrix} (\vec{x}_0, \vec{x}_1) & (\vec{x}_0, \vec{y}_1) & (\vec{x}_0, \vec{z}_1) \\ (\vec{y}_0, \vec{y}_1) & (\vec{y}_0, \vec{y}_1) & (\vec{y}_0, \vec{z}_1) \\ (\vec{z}_0, \vec{x}_1) & (\vec{z}_0, \vec{y}_1) & (\vec{z}_0, \vec{z}_1) \end{bmatrix} \cdot \begin{bmatrix} a_1 \\ b_1 \\ c_1 \end{bmatrix} \quad (3)$$

Maticový tvar rovnic (3) odpovídá transformaci souřadnic ze soustavy S_{x_1, y_1, z_1} do soustavy S_{x_0, y_0, z_0} . Vektor \vec{p} vyjádřený v soustavě S_{x_0, y_0, z_0} označíme \vec{p}_0 , v soustavě S_{x_1, y_1, z_1} ho označíme \vec{p}_1 a matici vektorových součinů nazveme maticí rotace R. Platí tedy:

$$\begin{aligned} \vec{p}_0 &= R \cdot \vec{p}_1 \\ \vec{p}_1 &= R^T \cdot \vec{p}_0 \\ R^T &= R^{-1} \end{aligned} \quad (4)$$

Skládání rotací

Dále si označíme matice R jako R_1^0 , která definuje orientaci SS S_{x_1, y_1, z_1} vzhledem k SS S_{x_0, y_0, z_0} . Podobně R_1^2 definuje orientaci SS S_{x_2, y_2, z_2} vzhledem k SS S_{x_1, y_1, z_1} . Platí tedy:

$$\begin{aligned} \vec{p}_0 &= R_1^0 \cdot \vec{p}_1 \\ \vec{p}_1 &= R_1^2 \cdot \vec{p}_2 \\ \vec{p}_0 &= R_2^0 \cdot \vec{p}_2 \end{aligned} \quad (5)$$

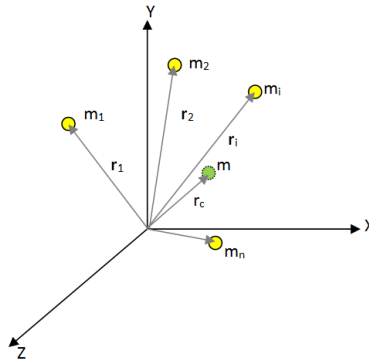
$$R_2^0 = R_1^0 \cdot R_1^2 \quad (6)$$

Matice R_2^0 definuje orientaci soustavy S_{x_2, y_2, z_2} vzhledem k SS S_{x_0, y_0, z_0}

2.2 Těžiště soustavy hmotných bodů

Máme-li soustavu hmotných bodů, je těžiště této soustavy bod, jehož hmotnost odpovídá součtu hmotností všech bodů soustavy, a jeho poloha je určena takto:

$$\begin{aligned} m &= \sum_{i=1}^n m_i \\ \mathbf{r}_c &= \frac{1}{m} \cdot \sum_{i=1}^n \mathbf{r}_i \cdot m_i \end{aligned} \quad (7)$$



Obrázek 3: Soustava hmotných bodů

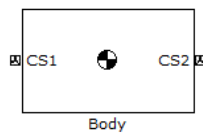
3 Model skákacího robotu v SimMechanics

3.1 Funkční bloky MATLAB/SIMULINK/SimMechanics

Prostředí MATLAB/SIMULINK/SimMechanics bylo navrženo pro modelování mechanických systémů pomocí jednoduchých funkčních bloků. Při vytváření jednoduchého i složitějšího modelu nepotřebujeme odvozovat matematický popis, stačí znát pouze strukturu modelu a vědět, jak tuto strukturu reprezentovat. Nyní popíšeme několik základních funkčních bloků, které použijeme pro modelování skákacího robotu.

Hmotný bod (BODY)

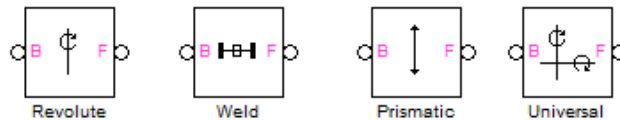
Hmotný bod je bod prostoru, kterému je přiřazena hmotnost o určité velikosti, ale je to bezrozměrný útvar. Nemá smysl uvažovat jeho deformaci nebo otáčení. V SimMechanics mu můžeme nastavit hmotnost a jeho umístění v prostoru. Více hmotných bodů můžeme spojovat pomocí vazeb (JOINTS).



Obrázek 4: Hmotný bod

Vazby mezi hmotnými body (JOINTS)

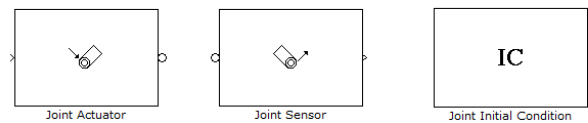
Vazby odpovídají různým typům připojení hmotných bodů k mechanickému prostředí nebo jinému hmotnému bodu. Prostředí SimMechanics rozlišuje pár desítek různých vazeb. V modelu robotu jsou použity některé z nich: *Revolute*, *Universal*, *Weld*, *Prismatic*. *Revolute* je kloub, který umožňuje rotaci tělesa kolem jedné osy rotace v prostoru, kterou můžeme libovolně definovat. *Universal* je kloub, který umožňuje rotaci kolem dvou os rotace současně. *Weld* představuje pevné spojení hmotných bodů bez možnosti translace nebo rotace. *Prismatic* je spojení hmotných bodů, které umožňuje translační pohyb v daném směru.



Obrázek 5: JOINTS

Sensors, Actuators

Pomocí *Joint Sensor* můžeme v průběhu simulace vyčítat různé veličiny vztahující se ke kloubu a na něj připojených těles. Například u kloubu *Revolute* můžeme zjistit při jeho rotaci během simulace aktuální úhel. Pomocí *Joint Actuator* můžeme u kloubu *Revolute* ovládat jeho natočení kolem nadefinované osy rotace. Blok *IC* představuje vložení počátečních podmínek, ve kterých se kloub nachází v nulovém čase simulace.

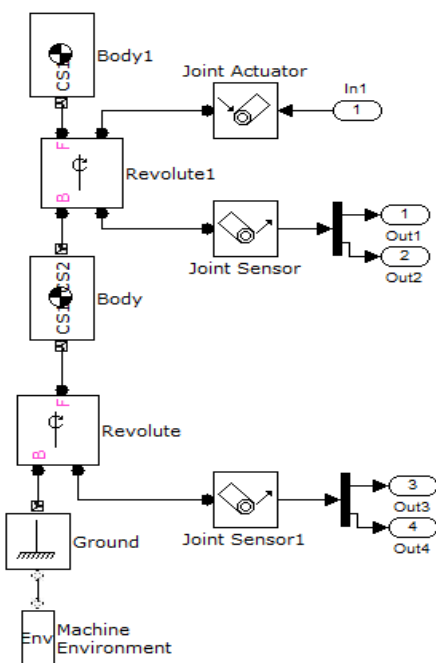


Obrázek 6: JOINTS

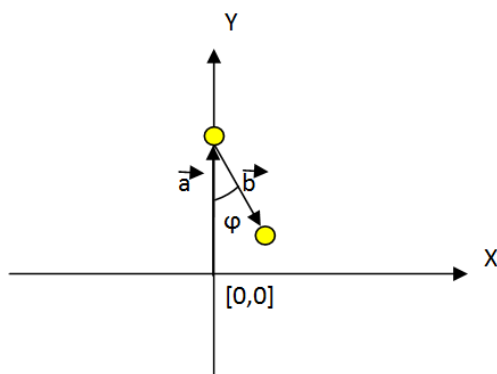
3.2 Dvojitě kyvadlo

Základní myšlenkou pro stabilizaci skákacího robotu je zvedání tří ramen tak, aby robot zůstal stát ve svislé poloze. Řízení těchto tří ramen v prostoru je poměrně složité, proto si nejdříve ověříme základní princip stabilizace na zjednodušeném modelu dvojitěho kyvadla. Kyvadla jsou umístěna v rovině

XY a mohou rotovat kolem osy Z. Dvojité kyvadlo reprezentují v prostředí SimMechanics takto:



Obrázek 7: Model dvojitého kyvadla v SimMechanics



Obrázek 8: Model dvojitého kyvadla v Kartézské soustavě souřadnic XY

3.3 Označení parametrů robotu

Model skákacího robotu se dá reprezentovat pomocí 6ti hmotných bodů, které umístíme do pravotočivé soustavy souřadnic dimenze 3 viz. Obrázek 1.

Máme tedy dány body O,S,E,A,B,C a jejich hmotnosti:

m hmotnost ramen(bodů A,B,C)

m_e hmotnost bodu E a O

m_s hmotnost bodu S

Vzdálenosti mezi body jsou označeny takto:

$$\|\vec{OS}\| = \frac{l}{2} \quad \|\vec{SA}\| = r \quad \|\vec{OE}\| = l \quad (8)$$

Dále označíme vektory s počátečním bodem v počátku kartézské SS^2 , kde je umístěn bod O, a koncovým bodem v daných bodech takto:

$$\vec{OS} = \vec{r}_s, \vec{OE} = \vec{r}_e, \vec{OA} = \vec{r}_a, \vec{OB} = \vec{r}_b, \vec{OC} = \vec{r}_c$$

Úhly $\varphi_A, \varphi_B, \varphi_C$ jsou úhly mezi těmito vektory:

\vec{SA} a \vec{SO} svírají úhel φ_A

\vec{SB} a \vec{SO} svírají úhel φ_B

\vec{SC} a \vec{SO} svírají úhel φ_C

Úhel δ je úhel, který svírají vektory $\vec{SA}, \vec{SB}, \vec{SC}$, promítneme-li je kolmo do roviny XZ. $\delta = 120^\circ$

3.4 Určení vektorů systému

Souřadnice vektoru \vec{r}_s, \vec{r}_e určíme velmi snadno pomocí daných parametrů:

$$r_s = \begin{bmatrix} 0 \\ \frac{l}{2} \\ 0 \end{bmatrix}, r_e = \begin{bmatrix} 0 \\ l \\ 0 \end{bmatrix}.$$

Vektor r_a určíme jako součet vektorů r_s a \vec{SA} . Vektory \vec{SO} a \vec{SA} mezi sebou svírají úhel φ_A , pomocí kterého určíme souřadnice vektoru \vec{SA} takto:

$$\vec{SA} = \begin{bmatrix} r \sin(\varphi_A) \\ -r \cos(\varphi_A) \\ 0 \end{bmatrix}$$

Předpokládáme, že bod A leží v rovině XY, proto je z-tová souřadnice rovna nule.

$$\vec{r}_a = \vec{r}_s + \vec{SA} = \begin{bmatrix} r \sin(\varphi_A) \\ \frac{l}{2} - r \cos(\varphi_A) \\ 0 \end{bmatrix}$$

Vektory \vec{r}_b, \vec{r}_c už tak snadno neurčíme, protože jejich z-tová souřadnice není

²Soustava souřadnic

nulová. Pro určení souřadnic využijeme **matici rotace** kolem osy Y o úhel $\delta = 120^\circ$. Matice rotace má tvar:

$$R_\delta = \begin{bmatrix} \cos \delta & 0 & -\sin \delta \\ 0 & 1 & 0 \\ \sin \delta & 0 & \cos \delta \end{bmatrix} = \begin{bmatrix} -\frac{1}{2} & 0 & -\frac{\sqrt{3}}{2} \\ 0 & 1 & 0 \\ \frac{\sqrt{3}}{2} & 0 & -\frac{1}{2} \end{bmatrix} \quad (9)$$

Využitím (9) tedy dostaneme:

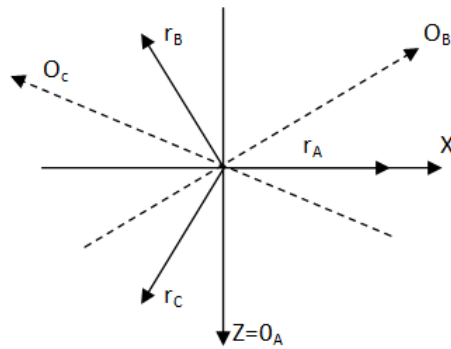
$$\vec{r}_b = R_\delta \cdot \vec{r}_a = \begin{bmatrix} -\frac{1}{2}r \sin(\varphi_B) \\ \frac{l}{2} - r \cos(\varphi_B) \\ \frac{\sqrt{3}}{2}r \sin(\varphi_B) \end{bmatrix} \quad (10)$$

$$\vec{r}_c = R_\delta \cdot \vec{r}_b = \begin{bmatrix} -\frac{1}{2}r \sin(\varphi_C) \\ \frac{l}{2} - r \cos(\varphi_C) \\ \frac{\sqrt{3}}{2}r \sin(\varphi_C) \end{bmatrix} \quad (11)$$

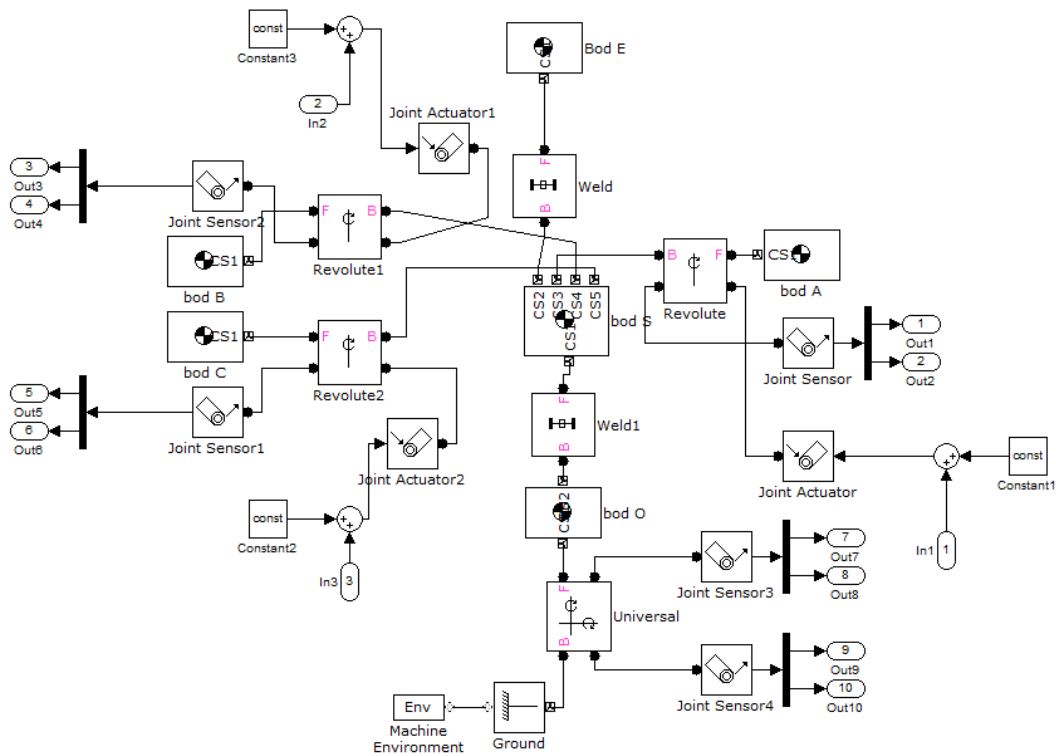
3.5 Skákací robot-varianta 1

Model skákacího robotu vytvoříme v SimMechanics obdobně jako model dvojitého kyvadla. Robot reprezentujeme pomocí 6ti hmotných bodů. Body O,S,E umístíme na osu Y a body A,B,C jsou koncové body ramen $\vec{r}_A, \vec{r}_B, \vec{r}_C$. Souřadnice všech bodů modelu jsme určili v předchozím odstavci. Potřebujeme tedy ještě určit osy otáčení jednotlivých ramen. Musíme dát pozor na správnou orientaci os, kterou určíme pravidlem pravé ruky. Osy rotace O_A, O_B, O_C mají tyto směrové vektory:

$$O_A : n_A = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \quad O_B : n_B = \begin{bmatrix} \sqrt{3} \\ 2 \\ 0 \\ -1 \\ -\frac{1}{2} \end{bmatrix} \quad O_C : n_C = \begin{bmatrix} -\sqrt{3} \\ -2 \\ 0 \\ -1 \\ -\frac{1}{2} \end{bmatrix} \quad (12)$$



Obrázek 9: Pravoúhlý průmět skákacího robotu do roviny XZ



Obrázek 10: Model skákacího robotu v SimMechanics - varianta 1

Tento model na obrázku použijeme pro linearizaci modelu při návrhu řízení. Pokud chceme, aby ramena mohla být zdvižena v určité poloze, musíme k řízení daných ramen přičíst moment síly, tak aby byl v rovnováze s účinky

gravitace. Moment síly zajistí, že ramena budou ve svislé poloze natočena podle úhlu φ . Moment síly se vypočte takto:

$$M_0 = r \cdot m \cdot g \cdot \sin(\varphi) \quad (13)$$

kde

g gravitační zrychlení

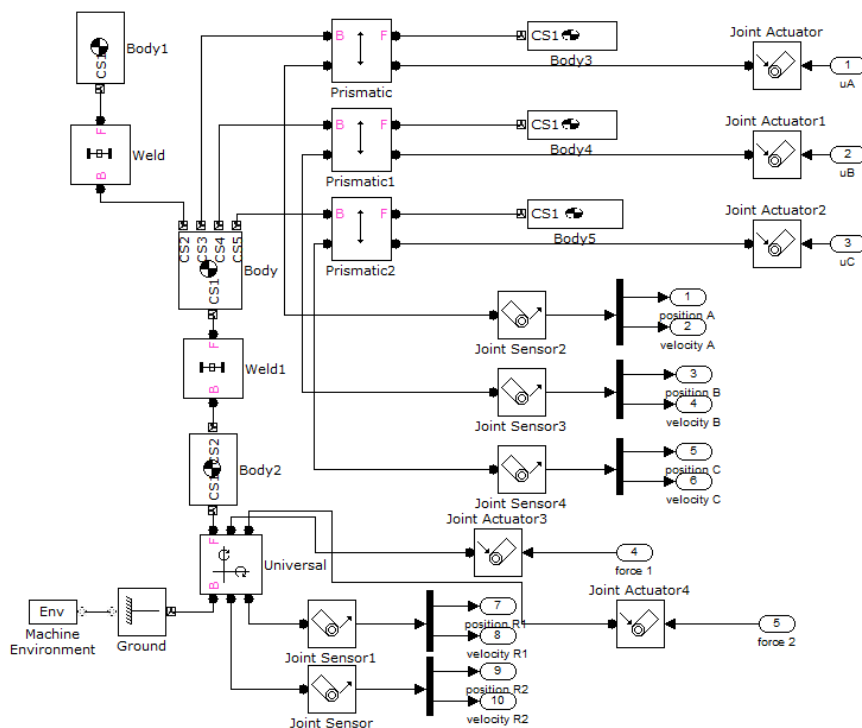
m hmotnost ramena

r délka ramena

φ úhel, ve kterém chceme systém linearizovat

3.6 Skákací robot - varianta 2

Druhá varianta pro vytvoření modelu skákacího robotu bude mít stejné parametry jako první varianta. Lišit se bude v tom, že ramena nebudou konat otáčivý pohyb. Na místo toho docílíme stabilizaci robotu translačním pohybem ramen. Ramena budou tři a jejich odchylka bude také 120° . Model v SimMechanicsu:



Obrázek 11: Model skákacího robotu v SimMechanics - varianta 2

4 Analýza modelu

4.1 Vyjádření obecné polohy těžiště systému

Abychom docílili vyjádření obecné polohy musíme nechat systém postupně rotovat kolem dvou os soustavy souřadnic. Jako osy rotace zvolíme osu Z a osu Y. Rotaci kolem osy Z o úhel ε vyjadřuje matice rotace R_ε a rotaci kolem osy Y vyjadřuje matice rotace R_δ . Matice mají tvar:

$$R_\delta = \begin{bmatrix} \cos \delta & 0 & -\sin \delta \\ 0 & 1 & 0 \\ \sin \delta & 0 & \cos \delta \end{bmatrix}$$

$$R_\varepsilon = \begin{bmatrix} \cos \varepsilon & -\sin \varepsilon & 0 \\ \sin \varepsilon & \cos \varepsilon & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad (14)$$

Nyní tedy máme SS S_{x_2, y_2, z_2} , která je pootočená vzhledem k SS S_{x_0, y_0, z_0} , ve které jsme si vyjádřili souřadnice vektorů všech hmotných bodů systému. Využijí tedy odvozený vztah (5),(6) a vyjádřím si souřadnice vektorů systému v S_{x_0, y_0, z_0} takto:

$$\begin{aligned} \vec{a} &= R_\delta \cdot R_\varepsilon \cdot \vec{r}_a \\ \vec{b} &= R_\delta \cdot R_\varepsilon \cdot \vec{r}_b \\ \vec{c} &= R_\delta \cdot R_\varepsilon \cdot \vec{r}_c \\ \vec{e} &= R_\delta \cdot R_\varepsilon \cdot \vec{r}_e \\ \vec{s} &= R_\delta \cdot R_\varepsilon \cdot \vec{r}_s \end{aligned} \quad (15)$$

$$\vec{a} = \begin{bmatrix} r \cos(\delta) \cos(\varepsilon) \sin(\varphi_A) - \cos(\delta) \sin(\varepsilon) \left(\frac{l}{2} - r \cos(\varphi_A) \right) \\ r \sin(\varepsilon) \sin(\varphi_A) + \cos(\varepsilon) \left(\frac{l}{2} - r \cos(\varphi_A) \right) \\ r \sin(\delta) \cos(\varepsilon) \sin(\varphi_A) - \sin(\delta) \sin(\varepsilon) \left(\frac{l}{2} - r \cos(\varphi_A) \right) \end{bmatrix}$$

$$\vec{b} = \begin{bmatrix} -\frac{r}{2} \cos(\delta) \cos(\varepsilon) \sin(\varphi_B) - \cos(\delta) \sin(\varepsilon) \left(\frac{l}{2} - r \cos(\varphi_B) \right) - \frac{\sqrt{3}r}{2} \sin(\delta) \sin(\varphi_B) \\ -\frac{r}{2} \sin(\varepsilon) \sin(\varphi_B) + \cos(\varepsilon) \left(\frac{l}{2} - r \cos(\varphi_B) \right) \\ -\frac{r}{2} \sin(\delta) \cos(\varepsilon) \sin(\varphi_B) - \sin(\delta) \sin(\varepsilon) \left(\frac{l}{2} - r \cos(\varphi_B) \right) + \frac{\sqrt{3}r}{2} \cos(\delta) \sin(\varphi_B) \end{bmatrix}$$

$$\vec{c} = \begin{bmatrix} -\frac{r}{2} \cos(\delta) \cos(\varepsilon) \sin(\varphi_C) - \cos(\delta) \sin(\varepsilon) \left(\frac{l}{2} - r \cos(\varphi_C) \right) + \frac{\sqrt{3}r}{2} \sin(\delta) \sin(\varphi_C) \\ -\frac{r}{2} \sin(\varepsilon) \sin(\varphi_C) + \cos(\varepsilon) \left(\frac{l}{2} - r \cos(\varphi_C) \right) \\ -\frac{r}{2} \sin(\delta) \cos(\varepsilon) \sin(\varphi_C) - \sin(\delta) \sin(\varepsilon) \left(\frac{l}{2} - r \cos(\varphi_C) \right) - \frac{\sqrt{3}r}{2} \cos(\delta) \sin(\varphi_C) \end{bmatrix}$$

$$\vec{e} = \begin{bmatrix} -l \cos(\delta) \sin(\varepsilon) \\ l \cos(\varepsilon) \\ -l \sin(\delta) \sin(\varepsilon) \end{bmatrix}$$

$$\vec{s} = \begin{bmatrix} -\frac{l}{2} \cos(\delta) \sin(\varepsilon) \\ \frac{l}{2} \cos(\varepsilon) \\ -\frac{l}{2} \sin(\delta) \sin(\varepsilon) \end{bmatrix}$$

K optimalizaci parametrů potřebujeme vyjádřit souřadnice polohy těžiště T. Abychom dokázali systém při vychýlení z rovnovážné polohy stabilizovat, musíme být schopni změnou úhlů φ_A , φ_B , φ_C dostat těžiště na osu Y. Nebo-li x-ová a z-tová souřadnice těžiště musí být nulové. Pro polohu těžiště platí:

$$\vec{t} = \frac{1}{m_c} \cdot \{m_s \cdot \vec{s} + m_e \cdot \vec{e} + m \cdot (\vec{a} + \vec{b} + \vec{c})\} \quad (16)$$

4.2 Optimalizace parametrů podle speciálního případu

Abychom optimalizovali parametry, vyjdeme z úvahy umístění x-vé a z-vé souřadnice těžiště nad počátek SS. Po roznásobení výpočtu souřadnic těžiště dostaneme velmi dlouhé rovnice, proto si zkráceně označíme goniometrické fce takto:

$$\cos(x) = c(x) \quad \sin(x) = s(x) \quad (17)$$

Za hmotnosti a délky dosadím:

$$m = \frac{1}{4} \quad m_s = \frac{1}{6} \quad m_e = \frac{1}{30} \quad l = 0.2 \quad r = 0.08 \quad (18)$$

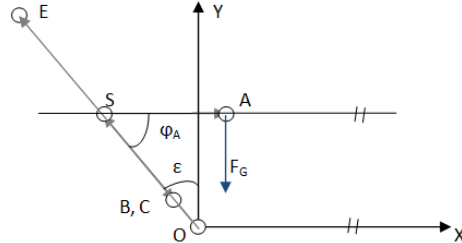
Pro x-ovou souřadnici těžiště pak platí:

$$\begin{aligned} t_x = 0 = & -\left(\frac{7}{295}\right)c(\delta)s(\varepsilon) + \left(\frac{6}{295}\right)c(\delta)c(\varepsilon)s(\varphi_A) - \left(\frac{15}{59}\right)c(\delta)s(\varepsilon)\left(\frac{1}{10} - \left(\frac{2}{25}\right)c(\varphi_A)\right) \\ & - \left(\frac{3}{295}\right)c(\delta)c(\varepsilon)s(\varphi_B) - \left(\frac{15}{59}\right)c(\delta)s(\varepsilon)\left(\frac{1}{10} - \left(\frac{2}{25}\right)c(\varphi_B)\right) - \left(\frac{3}{295}\right)s(\delta)\sqrt{3}s(\varphi_B) \\ & - \left(\frac{3}{295}\right)c(\delta)c(\varepsilon)s(\varphi_C) - \left(\frac{15}{59}\right)c(\delta)s(\varepsilon)\left(\frac{1}{10} - \left(\frac{2}{25}\right)c(\varphi_C)\right) + \left(\frac{3}{295}\right)s(\delta)\sqrt{3}s(\varphi_C) \end{aligned} \quad (19)$$

Pro z-ovou souřadnici těžiště pak platí:

$$\begin{aligned} t_z = 0 = & -\left(\frac{7}{295}\right)s(\delta)s(\varepsilon) + \left(\frac{6}{295}\right)s(\delta)c(\varepsilon)s(\varphi_A) - \left(\frac{15}{59}\right)s(\delta)s(\varepsilon)\left(\frac{1}{10} - \left(\frac{2}{25}\right)c(\varphi_A)\right) \\ & - \left(\frac{3}{295}\right)s(\delta)c(\varepsilon)s(\varphi_B) - \left(\frac{15}{59}\right)s(\delta)s(\varepsilon)\left(\frac{1}{10} - \left(\frac{2}{25}\right)c(\varphi_B)\right) + \left(\frac{3}{295}\right)c(\delta)\sqrt{3}s(\varphi_B) \\ & - \left(\frac{3}{295}\right)s(\delta)c(\varepsilon)s(\varphi_C) - \left(\frac{15}{59}\right)s(\delta)s(\varepsilon)\left(\frac{1}{10} - \left(\frac{2}{25}\right)c(\varphi_C)\right) - \left(\frac{3}{295}\right)c(\delta)\sqrt{3}s(\varphi_C) \end{aligned} \quad (20)$$

V dalším kroku si zjednodušíme rovnice (19) a (20) tak, že zvolíme $\delta = 0$. Pokud bude $\delta = 0$, potom se bude tělo robotu pohybovat pouze v rovině $XY \Rightarrow t_z = 0$. Vychýlení těla robotu vůči ose Y označíme ε .



Obrázek 12: Situace při maximální náklonu těla robotu

Budeme tedy zkoumat jaký maximální úhel ε pro různé parametry systému, můžeme dosáhnout. Největší účinek na robota bude mít tíhová síla F_G viz. Obrázek 12, pokud bude rameno robotu r_A rovnoběžné s osou X. Předpokládáme, že ramena r_B a r_C budou ležet v ose těla robotu, protože tím budou mít nejmenší účinek na odchýlení robotu od osy Y. Z těchto předpokladů tedy dostáváme, že pro úhly v modelu robotu platí:

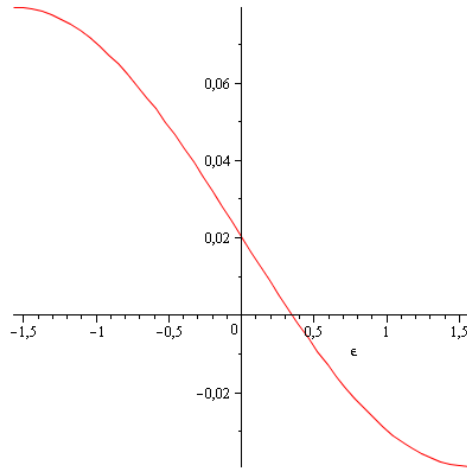
$$\varphi_A = \frac{\pi}{2} - \varepsilon \quad \varphi_B = \varphi_C = 0 \quad \delta = 0 \quad (21)$$

Použijeme tedy vztahy pro úhly (21) a dosadíme je do rovnice (19), čímž se vyjádření souřadnic těžiště robotu velice zjednoduší:

$$t_x = 0 = -\frac{2}{59} \sin(\varepsilon) + \frac{6}{295} \cos^2(\varepsilon) - \frac{2}{59} \sin(\varepsilon) \left(\frac{1}{10} - \frac{2}{25} \sin(\varepsilon) \right) \quad (22)$$

$$t_z = 0$$

Pokud tedy chceme vědět úhel ε vyřešíme rovnici (23). V intervalu $\langle 0, \frac{\pi}{2} \rangle$ nalezneme právě jedno řešení $\varepsilon = 20.05^\circ$.



Obrázek 13: Situace při maximální náklonu těla robotu

Druhá situace pro tento speciální případ je, že tělo robotu se odkloní od osy Y v opačném směru. Pak tedy natočíme rameno r_A do osy těla robotu a ramena r_B a r_C umístíme tak, aby leželi v rovině rovnoběžné s rovinou XZ. Pro úhly φ bude platit:

$$\varphi_A = 0 \quad \varphi_B = \varphi_C = \gamma \quad \delta = 0 \quad (23)$$

Pokud dosadíme tyto podmínky (23) do rovnic (19) a (20) dostaneme opět zjednodušené vztahy:

$$t_x = -\frac{17}{590} \sin(\varepsilon) - \frac{6}{295} \cos(\varepsilon) \sin(\gamma) - \frac{30}{59} \sin(\varepsilon) \left(\frac{1}{10} - \frac{2}{25} \cos(\gamma) \right) \quad (24)$$

$$t_z = 0$$

V intervalu $\langle 0, -\frac{\pi}{2} \rangle$ dostaneme řešení $\varepsilon = -13.6872^\circ$. Což ukazuje, že maximální náklon těla od osy Y v opačném směru je menší. Pokud měníme hmotnosti bodů systému a délky ramen robotu, rovnice (23) a (25) se pozmění a s nimi i jejich řešení. Parametry můžeme měnit tak, abychom dostali co největší úhel ε . Pokud takto měníme hmotnosti jednotlivých bodů a délky ramen dospějeme k závěru, že maximální odklon od osy Y nezávisí na jejich velikosti, ale jen v jakém poměru vzdálenosti a hmotnosti jednotlivých bodů zvolíme.

5 Návrh řídicího systému robotu

5.1 Návrh řízení dvojitého kyvadla

Nyní chceme docílit toho, aby se delší kyvadlo vrátilo do svislé polohy pro co možná největší počáteční úhel. Musíme tedy zvolit vhodný regulátor.

Pro tuto podmínku je vhodný stavový regulátor, který řídí počáteční stav $\mathbf{x}(0)$ "do nuly". Pro návrh stavového regulátoru je potřeba znát lineární stavový popis systému:

$$\begin{aligned}\dot{x} &= Ax + Bu \\ y &= Cx + Du\end{aligned}\tag{25}$$

Obecně bude systém dvou kyvadel nelineární. Pokud však vezmeme konkrétní bod³ stavového prostoru, můžeme chování systému linearizovat. Pro model kyvadel na obrázku (8) zvolíme jako "nulový stav" polohu, kde obě kyvadla \vec{a}, \vec{b} mají stejný směr jako osa Y a zároveň v ose Y leží, úhel $\varphi = 0$. Linearizaci provedu v Matlabu příkazem:

$$[\mathbf{A}, \mathbf{B}, \mathbf{C}, \mathbf{D}] = \text{linmod}(\text{jmeno modelu})\tag{26}$$

Příkazem (26) dostaneme stavový popis systému 4. řádu. Matice přímého působení vstupu na výstup bude nulová ($D=0$). Pro správnou funkci příkazu (26) je třeba označit pro model v SimMechanics vstupy a výstupy systému viz. obrázek 7. Pomocí bloků **Joint Sensor** a **Joint Actuator** získáme vstupy a výstupy (Out 1-4, In 1) systému. Výstupem systému je úhel a úhlová rychlost a vstupem systému je moment síly. Pro návrh stavového regulátoru musí být systém říditelný. Matice říditelnosti:

$$Q_r = [B \quad AB \quad A^2B \quad A^3B]\tag{27}$$

Systém bude říditelný v tom případě, pokud matice říditelnosti (27) bude mít plnou řádkovou hodnotu. Pro případ kyvadel musí tedy mít hodnotu 4. To ověříme v Matlabu příkazem:

$$\text{rank}(Q_r)\tag{28}$$

K linearizovanému popisu systému lze tedy navrhnout stavový regulátor. Rovnice stavového regulátoru:

$$\mathbf{u} = F\mathbf{C}\mathbf{x}\tag{29}$$

Stavové rovnice systému se změní na tvar ($D=0$) :

$$\begin{aligned}\dot{x} &= (A + BFC)x \\ y &= C^{-1}x\end{aligned}\tag{30}$$

Matici systému:

$$A + BFC\tag{31}$$

Chceme nalézt matici F takovou, aby matice systému byla podobná matici L. Pokud matice L je diagonální matice, pak prvky diagonály jsou vlastní

³bod vyadřuje stav systému

čísla matice (31). Z podobnosti matic (31) a L vyplývá že, musí existovat regulární matice T taková,že:

$$\begin{aligned} A + BFC &= TLT^{-1} \\ AT - TL + BFCT &= 0 \end{aligned} \quad (32)$$

Hledáme nyní řešení rovnice (32). Abychom našli regulární matici T musíme náhodně zvolit matici $H \in R^{m \times n}$ a matici L, kde n je počet stavů a m je počet vstupů. Pro matici H platí:

$$H = FCT \quad (33)$$

Po dosazení matice H do rovnice (32) platí:

$$AT - TL + BH = 0 \quad (34)$$

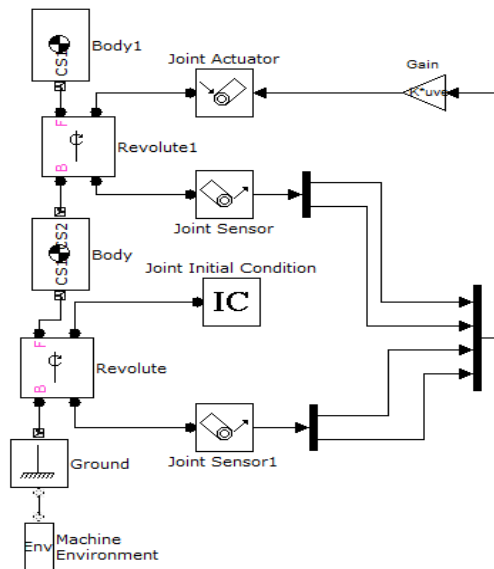
Maticovou rovnici (34) můžeme řešit v Matlabu pomocí příkazu:

$$T = \text{lyap}(A, -L, BH) \quad (35)$$

a potom dostaneme matici F:

$$F = HT^{-1}C^{-1} \quad (36)$$

Nyní můžeme uzavřít zpětnou vazbu a model v SIMULINKU vypadá takto:



Obrázek 14: Model v SimMechanics se zpětnou vazbou

Pokud systém uzavřeme zpětnou vazbou viz. obrázek (14), potom lze stabilizovat kyvadlo ve svislé poloze i když počáteční podmínky simulace nebudou nulové. Maximální výchylka vektoru \vec{a} od osy Y, pro kterou lze systém ustabilizovat, závisí na vhodné volbě vlastních čísel uzavřeného systému (prvky diagonály matice \mathbf{L}) a také na volbě parametrů systému (hmotnosti kyvadel a délky kyvadel).

5.2 Návrh stavového regulátoru podle kritéria ISE

Při návrhu regulátoru podle kritéria ISE chceme minimalizovat kvadratickou plochu regulační odchylky. Předpokládáme, že systém, pro který navrhujeme řízení je lineární. Obecně označujeme přenos regulátoru $F_r(p, \Theta)$, kde Θ jsou parametry regulátoru, které chceme určit tak, aby bylo minimalizováno kritérium ISE:

$$J(\Theta) = \int_0^{\infty} e^2(t, \Theta) dt \quad (37)$$

kde $e(t)$ je regulační odchylka. Uvažujeme stavový popis systému (25), kde matice přímého působení vstupu na výstup je nulová ($D=0$). Rovnice lineárního stavového regulátoru vypadá takto:

$$\mathbf{u}(t) = F\mathbf{x}(t) \quad (38)$$

Pro stavový regulátor má kritérium ISE tvar:

$$J(u) = \int_0^{\infty} (x^T Q x + u^T R u + 2x^T N u) dt \quad (39)$$

Pokud uvažujeme linearizovaný model robotu, který odpovídá systému 10. řádu se 3mi vstupy, jsou matice Q, N a R určeny takto:

$$Q = \begin{bmatrix} q_1 \\ q_1 \\ q_1 \\ q_2 \\ q_2 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \quad R = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \quad N = [0] \quad (40)$$

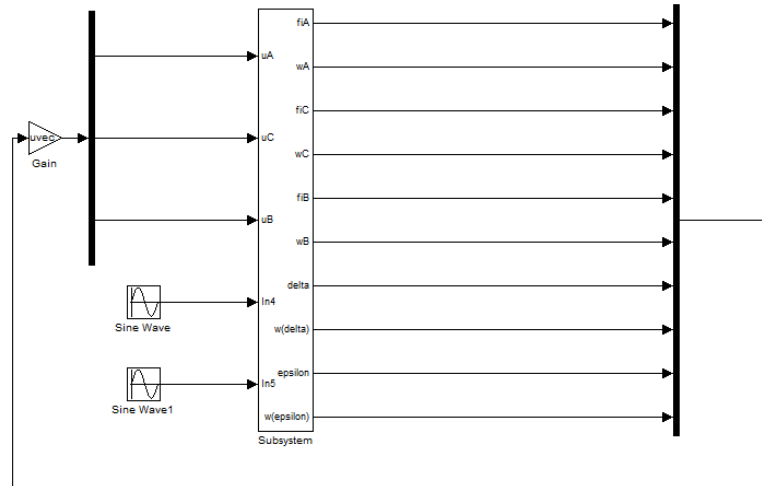
Parametry q_1, q_2 určují ztlumení jednotlivých stavů tak, abychom dosáhli lepších výsledků regulace. Pro výpočet minimalizace kritéria ISE použijeme v Matlabu příkaz:

$$[\mathbf{K}, \mathbf{S}, \mathbf{E}] = \text{lqr}(\mathbf{A}, \mathbf{B}, \mathbf{Q}, \mathbf{R}, \mathbf{N}) \quad (41)$$

Kde z matice K dopočteme zpětnovazební matici F takto:

$$F = -KC^{-1} \quad (42)$$

Nyní můžeme uzavřít regulační obvod. Maticí F přenásobíme měřené stavy a dostaneme tři vstupy systému. Regulační obvod tedy vypadá takto:



Obrázek 15: Regulační obvod

V regulačním obvodu je subsystém, který reprezentuje model robotu a je shodný buď s modelem na obrázku 10 nebo 11, záleží pro který systém řízení navrhujeme.

5.3 Stabilizace skákacího robotu v nakloněné poloze

Nyní chceme, aby se systém ustálil ve stavu, kdy tělo robotu bude odchýlené od osy Y o obecný úhel. Chceme tedy dopočítat úhly ramen a jejich řízení tak, aby se model ustálil v nakloněné poloze. Vyjdeme z předpokladu, že v ustáleném stavu pro stavovou rovnici platí:

$$\dot{\mathbf{x}} = 0 = \mathbf{Ax} + \mathbf{Bu} \quad (43)$$

Dále si rozdělíme vektor stavu na známou část a neznámou část, kterou chceme dopočítat. Vektor stavu rozdělíme na subvektory:

$$\mathbf{x} = \begin{bmatrix} x_2 \\ x_1 \\ \dot{x}_2 \\ \dot{x}_1 \end{bmatrix} \quad (44)$$

$x_1 \dots$ subvektor stavu, který obsahuje náklon těla robotu
 $x_2 \dots$ subvektor stavu, který obsahuje úhly ramen robotu

Nyní máme obecný vektor stavu rozdělený na subvektory a známe náklon těla robotu x_1^* . Pro ustálený stav platí:

$$0 = A \begin{bmatrix} x_2^* \\ x_1 \\ 0 \\ 0 \end{bmatrix} + Bu \quad (45)$$

Matici systému a vstupu rozdělíme na submatice:

$$\begin{bmatrix} A_{11} & A_{12} & A_{13} & A_{14} \\ A_{21} & A_{22} & A_{23} & A_{24} \\ A_{31} & A_{32} & A_{33} & A_{34} \\ A_{41} & A_{42} & A_{43} & A_{44} \end{bmatrix} \cdot \begin{bmatrix} x_2^* \\ x_1 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} B_1 \\ B_2 \\ B_3 \\ B_4 \end{bmatrix} \cdot u = 0 \quad (46)$$

Zjednodušíme rovnici (46) a dostaneme soustavu rovnic:

$$\begin{aligned} A_{11}x_2^* + A_{12}x_1 + B_1u &= 0 \\ A_{21}x_2^* + A_{22}x_1 + B_2u &= 0 \\ A_{31}x_2^* + A_{32}x_1 + B_3u &= 0 \\ A_{41}x_2^* + A_{42}x_1 + B_4u &= 0 \end{aligned} \quad (47)$$

Z rovnic (47) určíme x_2 a u . Obecně bude mnoho řešení x_2^*, u^* , my chceme ale vybrat řešení, kde bude řízení u minimální:

$$\|u\|_\infty \rightarrow \min \quad (48)$$

Pro výpočet požadovaného řešení x_2^*, u^* použijeme metodu nejmenších čtverců, kterou aplikujeme na rovnici:

$$P \cdot v = Q \quad (49)$$

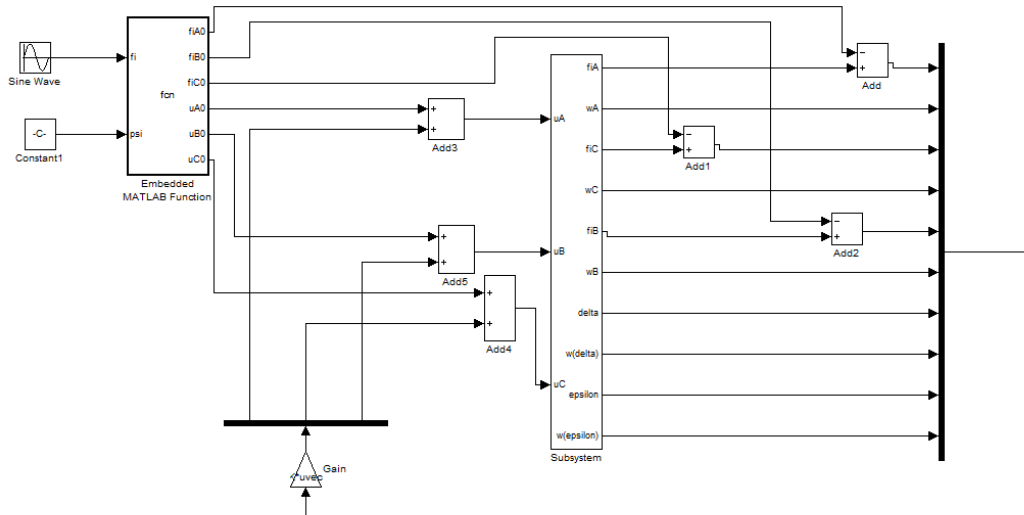
$$\underbrace{\begin{bmatrix} A_{11} & B_1 \\ A_{21} & B_2 \\ A_{31} & B_3 \\ A_{41} & B_4 \end{bmatrix}}_P \cdot \underbrace{\begin{bmatrix} x_2^* \\ u^* \end{bmatrix}}_v = \underbrace{\begin{bmatrix} -A_{12} \\ -A_{22} \\ -A_{32} \\ -A_{42} \end{bmatrix}}_Q \cdot x_1^*$$

Řešení rovnice (49) metodou nejmenších čtverců dostaneme v Matlabu pomocí příkazu:

$$v = P \backslash Q \quad (50)$$

Z tohoto řešení tedy zjistíme jaké bude natočení ramen robotu v požadované poloze a jakým minimálním řízením se můžeme do této polohy dostat. Pokud

budeme chtít požadované natočení robotu během simulace měnit musíme průběžně počítat příslušné natočení ramen a minimální řízení. Regulační smyčka tedy bude vypadat takto:



Obrázek 16: Regulační obvod pro libovolné natočení robotu

6 Ověření funkčních vlastností

6.1 Simulace řízených kyvadel

Nyní si zvolíme parametry pro simulaci dvojitého kyvadla takto:

$$m = 1 \text{ kg} \quad m1 = 0.4 \text{ kg} \quad \|\vec{a}\| = 0.2 \text{ m} \quad \|\vec{b}\| = 0.09 \text{ m}$$

Pomocí matice L přiřadíme uzavřenému systému vlastní čísla. Matice L vypadá takto:

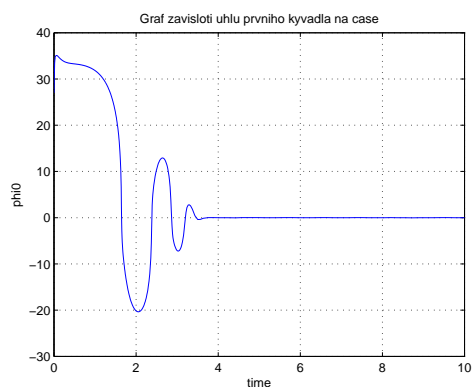
$$L = \begin{bmatrix} -\lambda & \lambda & 0 & 0 \\ -\lambda & -\lambda & 0 & 0 \\ 0 & 0 & -\lambda & \frac{\lambda}{2} \\ 0 & 0 & -\frac{\lambda}{2} & -\lambda \end{bmatrix} \quad (51)$$

Pro různou volbu parametru λ ověříme simulací, pro jaký maximální úhel ϕ^4 dokáže stavový regulátor ještě systém stabilizovat. Nejlepších výsledků pro danou matici L dosáhl systém při $\lambda = 14$. Maximální počáteční výchylka,

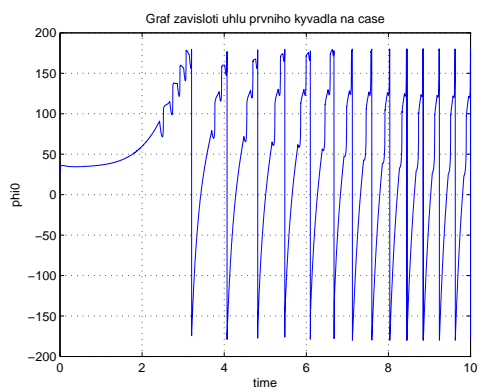
⁴úhel, který svírá \vec{a} s osou Y

kdy je regulátor schopný systém stabilizovat, je $\phi_0 = 27^\circ$. Pro úhel $\phi_0 = 28^\circ$ už stavový regulátor nedokáže systém stabilizovat.

Výsledky simulace:



Obrázek 17: Regulace při $\phi = 27^\circ$



Obrázek 18: Regulace při $\phi = 28^\circ$

6.2 Simulace stabilizace robotu

Zvolíme tedy parametry skákacího robotu takto:

$$l = 0.2 \text{ m} \quad r = 0.09 \text{ m} \quad m = 1/4 \text{ kg} \quad m_s = 1/6 \text{ kg} \quad m_e = 1/30 \text{ kg} \quad (52)$$

Uvažujeme-li pro model variantu 1, můžeme pro ramena robotu nadefinovat různou počáteční polohu. Prakticky si můžeme vybrat libovolný počáteční úhel $\varphi_A = \varphi_B = \varphi_C = \varphi$ a jsme schopni navrhnout řízení, které pro určitou výchylku model stabilizuje. Zajímavé je, že i v poloze kdy $\varphi = \frac{\pi}{2}$ nemá tíhová

síla při malé změně $\varphi_A = \varphi_B = \varphi_C$ téměř žádné účinky a přesto je robot schopný se stabilizovat. V tomto případě mají na tělo robotu větší účinky reaktivní síly způsobené otáčením ramen. Právě vhodná volba počátečního úhlu φ závisí na tom, o kolik stupňů maximálně můžeme tělo robotu vychýlit ze svislé polohy. Úhel δ je odchýlení při rotaci kolem osy Z a ε je odchýlení při rotaci kolem osy X. Pokud vezmeme linearizační polohu ramen $\varphi = \frac{\pi}{2}$ jsou maximální vychýlení těla robotu tyto:

$\delta[^\circ]$	$\varepsilon[^\circ]$
4.8	0
1	1
0	2.1

Asi nejlepší linearizační poloha je když $\varphi = \frac{\pi}{4}$ dosáhneme pak výsledků:

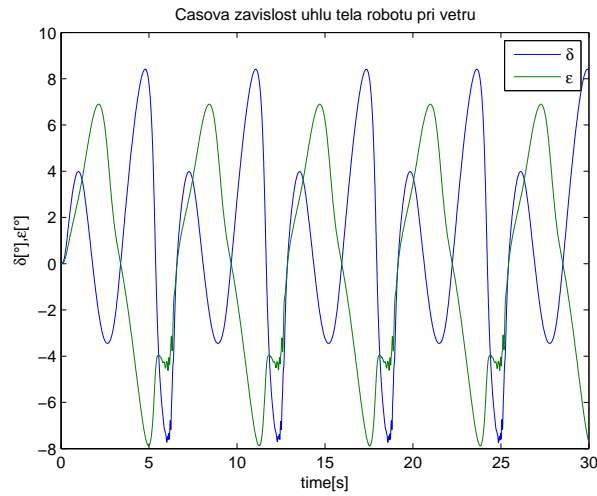
$\delta[^\circ]$	$\varepsilon[^\circ]$
18	0
16	16
0	19

Vezmeme další simulační případ, kdy na tělo robotu působí proměnná síla. Tato síla bude složená ze 2 sinusových signálů, kde jeden bude působit na rotaci kolem osy Z a druhý kolem osy Y. Signálům přiřadíme stejnou amplitudu, ale rozdílnou frekvenci. Signály mají tvar:

$$A_1 \sin(\omega_1 t)$$

$$A_2 \sin(\omega_2 t)$$

Tento simulační případ odpovídá realitě, kdy by naklon těla robotu ovlivňovalo foukání větru. Pro amplitudy $A_1 = A_2 = 0.03 \text{ N}$ a frekvence $\omega_1 = 5 \text{ rad} \cdot \text{s}^{-1}$ $\omega_2 = 6 \text{ rad} \cdot \text{s}^{-1}$ se dokáže systém robotu ve větru ještě udržet, pro amplitudu $A_1, A_2 \geq 0.04$ už podlehne větru a spadne. Výsledek simulace:



Obrázek 19: Graf závislosti vychýlení těla robotu δ , ε v čase při působení síly, modrý průběh je časová závislost δ , zelený graf je časová závislost ε

6.3 Libovolné naklonění těla robotu

Pomocí vychylování ramen můžeme tělo robotu naklonit, do kterékoliv polohy za podmínky, že je možné robota vychýlením ramen v této poloze stabilizovat. Tento předpoklad ověříme simulací. Chceme tedy, aby se úhel δ spojitě měnil např. podle signálu sinus s amplitudou $A = \frac{3 \cdot \pi}{180}$ a frekvencí $\omega = 1 \text{ rad} \cdot \text{s}^{-1}$.



Obrázek 20: Graf závislosti úhlu δ na čase

Jak je vidět z grafu, úhel se nemění úplně přesně podle signálu sinus, ale přibližně se mu podobá. Chyba je způsobená tím, že pokud se ramena natočí, nějakou chvíli trvá, než se tělo robotu do polohy dostane.

7 Návrh funkčního vzorku

K návrhu funkčního vzorku se dají použít různé technické součástky. Asi nejdůležitější součástí by měli být aktuátory, které použijeme na manipulaci s rameny robotu. Také bychom chtěli docílit toho, aby skákací robot byl přiměřeně malé velikosti.

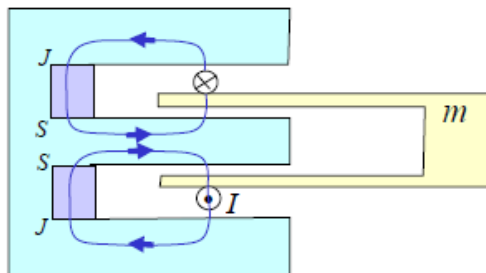
Pokud budeme konstruovat návrh robotu, půjde lépe realizovat model robotu pro variantu 2. Kde se ramena robotu pohybují po přímce. Pro malé rozměry je vhodné použít motory na principu "Voice coil" actuátoru nebo piezo actuátoru.

7.1 "Voice coil" actuátor

Voice coil actuátor funguje na indukčním principu. Elektro-magnetická indukce je jev, kdy vzniká elektrické napětí U_{ind} v uzavřeném elektrickém obvodu, který je způsoben změnou magnetického indukčního toku ϕ . Tento jev matematicky popisuje Faradayův zákon:

$$U_{ind} = \frac{d\phi}{dt} \quad (53)$$

V důsledku indukovaného napětí může ve vodiči protékat elektrický proud, který má podle Lenzova pravidla takový směr, že působí proti změně, která ho vyvolala. Máme tedy magnet a cívku:



Obrázek 21: Cívka zasunutá v magnetu

Změníme-li magnetický indukční tok působí na jeden závit síla:

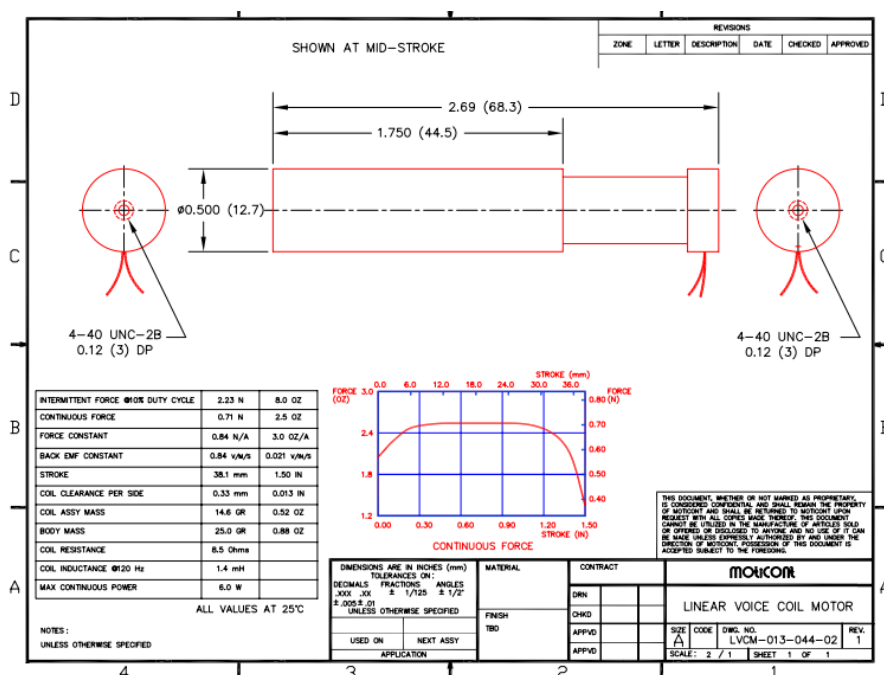
$$F = \oint I \times Bdl \quad (54)$$

Tato síla způsobí pohyb cívky v magnetickém poli.

Jako motor pro pohon ramen skákacího robotu by se dal použít "Voice coil" motor LVCM-013-044-02:



Obrázek 22: "Voice coil" motor

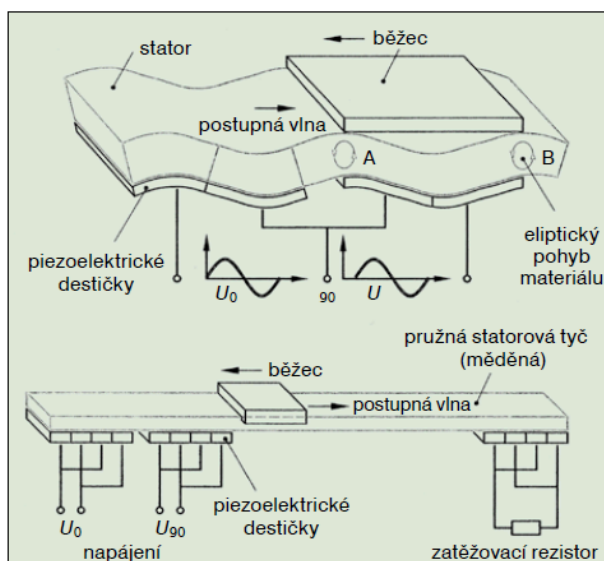


Obrázek 23: technický výkres s parametry "Voice coil" motoru

Pokud bychom chtěli koupit tento motor zaplatíme za jeden kus \$213.04. Tato cena platí při koupi 3 kusů motorů. Motory lze zakoupit v internetovém obchodu na stránkách viz. [4].

7.2 Piezo actuátor

Piezo motory jsou založeny na piezo elektrickém jevu. Tento jev je schopnost krystalu generovat elektrické napětí při jeho deformování, tato změna je vratná. U piezo motorů se používá opačný princip jevu, kdy po přivedení napětí na svorky motoru dojde k deformaci krystalové mřížky. Nejznámější piezo krystalovou látkou je křemen. Princip piezomotoru lze snadno vysvětlit na jeho lineární variantě:



Obrázek 24: Lineární piezo motor

” Pohyblivá část motoru (”běžec”) se pohybuje po vlnící se statorové tyči jako ”surfovací prkno” po příbojových vlnách. Stator lineárního piezomotoru je tvořen pružným měděným pásem, k jehož spodní straně jsou přilepeny piezoelektrické destičky. Zde nastává přeměna elektrické energie na mechanickou, vzniklými kmity se podél statorové tyče vytvoří mechanické kmity a postupná vlna, po které se v důsledku tření pohybuje ”běžec”.⁵

Vhodným piezo motorem pro realizaci ramen skákacího robotu by mohl být: *U-264 RodDrive Piezomotor Direct Drive*.

⁵viz. seznam zdrojů [3]



Obrázek 25: Lineární piezo motor

Parametry motoru U-264:

rychlost do 250 mm/s

rozsah pohybu 150 mm

tažný moment 15 N

8 Závěr:

Nakonec můžeme zhodnotit, do jaké míry se povedlo splnit zadání práce. Na začátku práce bylo třeba vytvořit idealizovaný model skákacího robota a matematický popis v soustavě souřadnic. Tento krok byl nezbytný k provedení analýzy systému a vytvoření modelu v programovém prostředí MATLAB/SIMULINK/ SimMechanics. V prostředí SimMechanics se povedlo namodelovat tento mechanický systém. Pro lepší analýzu modelu bychom mohli sestavit úplný matematický popis ve formě diferenciálních rovnic. K odvození pohybových rovnic by bylo vhodné použít Lagrangeovu metodu.

Uvažovali jsme dvě možné varianty, jak realizovat připojení ramen k robotu. Pro obě varianty se k modelu systému podařilo navrhnout řízení, pomocí kterého se dokáže robot stabilizovat ve svislé poloze. Pro obě varianty robot dokáže stát i když na něj působí vnější síla. Pro první variantu lze na tělo robot působit větší silou než u druhé, ale druhá varianta by šla lépe konstruovat v reálném případě.

Další problém, který je v práci řešený, je stabilizace robota v nakloněné poloze. Dosáhli jsme toho, že robot můžeme libovolně naklánět pomocí měnící se polohy ramen. Naklánění robota a jeho stabilizace v určité poloze jsou důležité v myšlence skákání robota do dálky. Při řízení robota do nakloněné polohy, ho nelze naklonit pro větší úhel než je 5° . Tato možnost malého náklonu může být způsobená omezenou platností popisu lineárního systému, pro který bylo řízení navrženo. Pokud navrhujeme řízení pro linearizovaný systém ve svislé poloze, lze očekávat pouze omezenou platnost této linearizace. Neplatnost linearizace způsobí to, že řízení se nebude vzhledem k aktuální poloze systému chovat správně. Pokud bychom tedy chtěli zvětšit maximální výchylku, do které můžeme robota pomocí řízení ramen naklonit, museli bychom použít metodu, která by linearizovala systém během simulace a následně přepočítávala řízení.

8.1 Seznam zdrojů:

- [1] Spong M.W., Robot Modeling and Control, John Wiley and Sons. INC.
- [2] Piezo Mechanisms Piezo Mechanic Systems for High Tech Applications [online]. 2009 [cit. 2012-05-15]. Dostupné z: <http://www.piezo.ws/index.php>
- [3] Prof. Václav ČERNÝ, II. Piezoelektrické motory – regulace a řízení. [online]. [cit. 2012-05-15]. Dostupné z: <http://www.odbornecasopisy.cz/download/el010408.pdf>
- [4] MontiCont a motion company [online]. 2008 [cit. 2012-05-15]. Dostupné z: <http://www.pwr-con.com/voice-coil-motor.htm>
- [5] A study of a jumping one-leg robot with two degrees of freedom ISAROB

2009, S.M.Najib,S.Yusoh, Shingo Yamashita, Naomi Kokubo, Yoshihiko Nomura.

[6] Dynamic Modeling and Control of Human Inspired Jumping Robot. Ljubljana, Slovenia, 2007. Article. Automatics, Biocybernetics and Robotics, Jožef Stefan Institute.