

МАТЕМАТИЧКА ГИМНАЗИЈА

МАТУРСКИ РАД
- из физике -

Симулација самобалансирајућег робота

Ученик:
Валентина Њаради IVд

Ментор:
Јовица Милисављевић

Београд, јун 2019.

Садржај

1	Увод	1
2	Моделирање система	3
2.1	Динамика робота	3
2.1.1	Линеаризација	6
2.2	Динамика мотора	6
2.3	Динамички модел система	8
2.4	Сензор	9
2.4.1	Акцелерометар	10
2.4.2	Жироскоп	10
2.4.3	Моделирање сензора	10
2.5	Систем за управљање	11
2.5.1	Филтри	11
2.5.2	Стабилизатори	15
3	Симулација	19
4	Закључак	23
	Литература	23

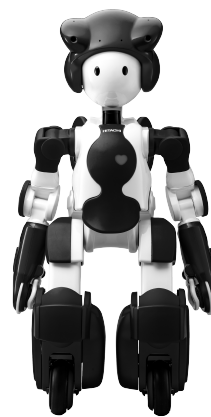
1

Увод

Проблем контроле самобалансирајућег робота на два точка је, због својих широких примена, веома популаран у последњих неколико деценија. Главни циљ многим научницима који су се бавили овим проблемом јесте налажење идеалног математичког модела, структурних карактеристика и алгоритама стабилизације робота. Неке од практичних примена робота који раде по овом принципу су превозна средства или аутоматска инвалидска колица. Сегвеј (*Segway*) роботи су најпознатији примери робота на два точка, са разним употребама као што су транспорт по унутрашњем и спољашњем простору, индустријска аутоматизација, системи одбране и лична превозна средства. Неки од познатих комерцијалних примера самобалансирајућих робота су *Segway Personal Transporter* - прво комерцијално доступно превозно средство овог типа, приказан на слици 1.1 и *Emiew* - хуманоидни робот на два точка компаније Хитачи, приказан на слици 1.2.



Слика 1.1: *Segway*



Слика 1.2: *Emiew*

Самобалансирајући робот се може представити као покретно инверзно клатно. Циљ овог рада је моделирање и симулација овог система. Изведене су једначине кретања инверзног клатна и једначине које описују мотор, од чега је добијем прецизан математички модел система. Описани су сензори помоћу којих се добија информација о углу отклона робота, као и њихове карактеристике. У поглављу 2.5.1 детаљно је описан Калманов филтар и комплементарни филтар који се користе за смањивање шума сигнала са сензора. Симулација је прављена са два различита стабилизатора: ПИД контролером и ЛКР контролером, који су описани у поглављу 2.5.2. Симулација је рађена у програмском пакету *MATLAB Simulink*. У трећем поглављу приказани су резултати направљене симулације.

2

Моделирање система

2.1 Динамика робота

Механички систем састоји се из тела робота и два точка. Тело се може моделовати као инверзно клатно где се центар масе налази изнад осе ротације. За извођење једначина посматрано је кретање само по хоризонталној оси и претпостављено је да нема клизања између точка и подлоге. Коришћен је Лагранжов метод за описивање динамике система [2, 3].

За описивање динамике система неопходно је дефинисати координате робота који ће бити посматрани као релевантни. То су координате центра масе (x_c, z_c) и центра точка (x) , угаони померај точка (θ) и угао између вертикале и праве која спаја центар точка са центром масе (φ) . Координате су обележене на слици 2.1. Параметри робота који ће бити коришћени за описивање координата система су: маса тела робота m , маса точка m_t , полупречник точка R , растојање између центра точка и центра масе L , момент инерције тела I и момент инерције точка I_t .

Можемо изразити $x, x_c, z_c, \dot{x}, \dot{x}_c$ и \dot{z}_c као:

$$x = R\varphi \tag{2.1}$$

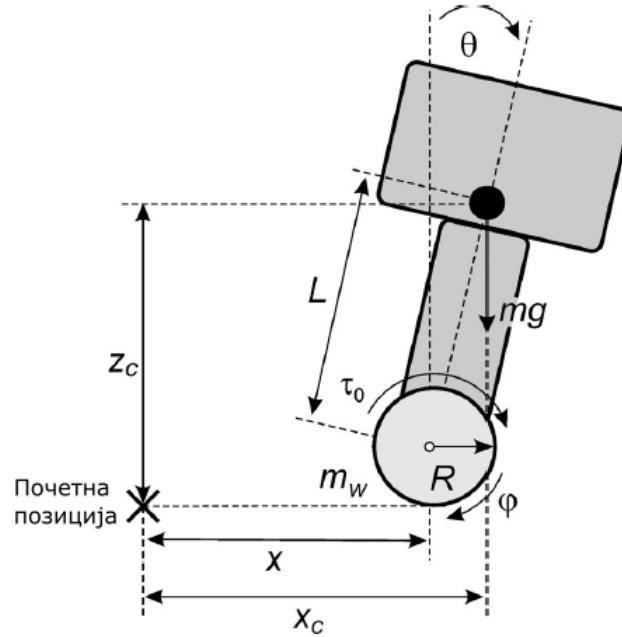
$$x_c = R\varphi + L\sin\theta \tag{2.2}$$

$$z_c = R + L\cos\theta \tag{2.3}$$

$$\dot{x} = R\dot{\varphi} \tag{2.4}$$

$$\dot{x}_c = R\dot{\varphi} + L\dot{\theta}\cos\theta \tag{2.5}$$

$$\dot{z}_c = -L\dot{\theta}\sin\theta \tag{2.6}$$



Слика 2.1: Физички модел самобалансирајућег робота

Даље, можемо дефинисати потенцијалну енергију (чија је вредност 0 у вертикалном положају) и кинетичку енергију као:

$$E_p = mg(R + L\cos\theta) - mg(R + L) = mgL(\cos\theta - 1) \quad (2.7)$$

$$E_k = 2\frac{1}{2}m_t\dot{x}^2 + 2\frac{1}{2}I_t\dot{\varphi}^2 + \frac{1}{2}m(\dot{x}_c^2 + \dot{z}_c^2) + \frac{1}{2}I\dot{\theta}^2 \quad (2.8)$$

Убацивши једначине (2.4), (2.5), (2.6) у једначину (2.8) добија се:

$$E_k = \frac{1}{2}(2I_t + 2m_tR^2 + mR^2)\dot{\varphi}^2 + \frac{1}{2}(I + mL^2)\dot{\theta}^2 + mRL\dot{\varphi}\dot{\theta}\cos\theta \quad (2.9)$$

Сума кинетичке и потенцијалне енергије представља укупну, тј. механичку енергију, док разлика ове две физичке величине представља Лагранжеву енергију L :

$$L = E_k - E_p = \frac{1}{2}(2I_t + 2m_tR^2 + mR^2)\dot{\varphi}^2 + \frac{1}{2}(I + mL^2)\dot{\theta}^2 + mRL\dot{\varphi}\dot{\theta}\cos\theta - mgL(\cos\theta - 1) \quad (2.10)$$

Лагранжева једначина друге врсте је дефинисана као:

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{q}_i} \right) - \frac{\partial L}{\partial q_i} = Q_i \quad (2.11)$$

где је q_i координата система, а Q_i укупан моменат који утиче на q_i . Једначине динамике за φ :

$$\frac{\partial L}{\partial \dot{\varphi}} = (2I_t + 2m_t R^2 + mR^2)\dot{\varphi} + mRL\dot{\theta}\cos\theta \quad (2.12)$$

$$\frac{\partial L}{\partial \varphi} = 0 \quad (2.13)$$

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\varphi}} \right) - \frac{\partial L}{\partial \varphi} = (2I_t + 2m_t R^2 + mR^2)\ddot{\varphi} + mRL\ddot{\theta}\cos\theta - mRL\dot{\theta}^2\sin\theta = \mu \quad (2.14)$$

где је μ укупан момент који делује на φ . Једначине динамике за θ :

$$\frac{\partial L}{\partial \dot{\theta}} = (I + mL^2)\dot{\theta} + mRL\dot{\varphi}\cos\theta \quad (2.15)$$

$$\frac{\partial L}{\partial \theta} = -mRL\dot{\varphi}\sin\theta + mgL\sin\theta \quad (2.16)$$

$$\frac{d}{dt} \left(\frac{\partial L}{\partial \dot{\theta}} \right) - \frac{\partial L}{\partial \theta} = (I + mL^2)\ddot{\theta} + mRL\ddot{\varphi}\cos\theta - mgL\sin\theta = \eta \quad (2.17)$$

где је η укупан момент који делује на θ .

Ово се у матричном облику може записати као:

$$\begin{bmatrix} 2I_t + 2m_t R^2 + mR^2 & mRL\cos\theta \\ mRL\cos\theta & I + mL^2 \end{bmatrix} \begin{bmatrix} \ddot{\varphi} \\ \ddot{\theta} \end{bmatrix} + \begin{bmatrix} 0 & -mRL\dot{\theta}\sin\theta \\ 0 & 0 \end{bmatrix} \begin{bmatrix} \dot{\varphi} \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ -mgL\sin\theta \end{bmatrix} = \begin{bmatrix} \mu \\ \eta \end{bmatrix} \quad (2.18)$$

Сада је потребно укупне моменте μ и η изразити преко познатих параметара. Укупни момент добијамо сабирањем обртног момента M_0 , момента силе трења између подлоге и тачкова M_{tr} и момента силе трења осовине M_{os} .

$$\mu = 2M_0 - M_{os} - M_{tr} = M_0 - \beta_{os}(\dot{\varphi} - \dot{\theta}) - \beta_{tr}\dot{\varphi} \quad (2.19)$$

$$\eta = -2M_0 + M_{os} = -M_0 + \beta_{os}(\dot{\varphi} - \dot{\theta}) \quad (2.20)$$

Ове релације се могу записати матрично и то као:

$$\begin{bmatrix} \mu \\ \eta \end{bmatrix} = \begin{bmatrix} 2 \\ -2 \end{bmatrix} M_0 - \begin{bmatrix} \beta_{os} + \beta_{tr} & -\beta_{os} \\ -\beta_{os} & \beta_{os} \end{bmatrix} \begin{bmatrix} \dot{\varphi} \\ \dot{\theta} \end{bmatrix} \quad (2.21)$$

2.1.1 Линеаризација

Како би даље решавање овог система било могуће, морају се увести апроксимације које ће важити у опсегу за који се креира контролер. Робот треба врло мало да осцилује, стога се може рећи да је угао отклоне θ јако мали. Због овога важи:

$$\cos\theta \approx 1, \sin\theta \approx \theta, \theta^2 \approx 0 \quad (2.22)$$

Користећи ове апроксимације, добија се линеаризован динамички систем:

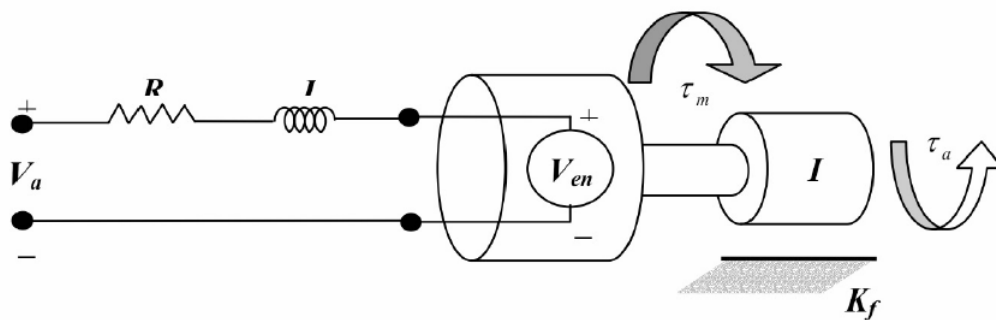
$$\begin{bmatrix} 2I_t + 2m_t R^2 + mR^2 & mRL\cos\theta \\ mRL\cos\theta & I + mL^2 \end{bmatrix} \begin{bmatrix} \ddot{\varphi} \\ \ddot{\theta} \end{bmatrix} + \begin{bmatrix} \beta_{os} + \beta_{tr} & -\beta_{os} \\ -\beta_{os} & \beta_{os} \end{bmatrix} \begin{bmatrix} \dot{\varphi} \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ -mgL \end{bmatrix} \theta = \begin{bmatrix} 2 \\ -2 \end{bmatrix} M_0 \quad (2.23)$$

Краће записано:

$$P \begin{bmatrix} \ddot{\varphi} \\ \ddot{\theta} \end{bmatrix} + Q \begin{bmatrix} \dot{\varphi} \\ \dot{\theta} \end{bmatrix} + R\theta = SM_0 \quad (2.24)$$

2.2 Динамика мотора

Из описане динамике система види се да је за израчунавање нагибног угла робота у датом тренутку потребан улазни момент силе. Пошто микроконтролеру није могуће ”рећи” да мотор направи одређени момент, потребно је израчунати како тај момент силе зависи од напона који микроконтролер саопштава мотору. Блок шема мотора приказана је на слици 2.2.



Слика 2.2: Блок шема мотора

Момент силе је сразмеран промени угаоне брзине мотора $\dot{\omega}$:

$$M_0 = I_t \dot{\omega} \quad (2.25)$$

Мотор производи момент τ ако се на њега доводи напон V_0 који је директно сразмеран струји i која пролази кроз мотор и јачине магнетног поља које стварају калемови у мотору. Претпоставља се да је јачина магнетног поља константна. У том случају момент мотора је директно пропорционалан струји са коефицијентом k_i :

$$\tau = k_i i \quad (2.26)$$

Пошто се ротор мотора окреће у магнетном пољу, индукује се електромоторна сила V_e која је директно сразмерна угаоној брзини окретања ротора мотора ω па важи:

$$V_e = k_e \omega \quad (2.27)$$

Мотор ће бити моделиран као редна веза отпорника и индуктора те се може написати II Кирхофов закон за то коло:

$$V_0 = iR + L \frac{di}{dt} + V_e \quad (2.28)$$

У једначини кретања мотора треће осовине апроксимира се као линеарна функција угаоне брзине ротора мотора са коефицијентом b . Према Њутновим законима кретања добија се:

$$\tau = M_0 + b\omega + I_M \dot{\omega} \quad (2.29)$$

где је I_M момент инерције мотора. За мале ДЦ моторе L је веома мало, па се компонента $L \frac{di}{dt}$ сматра занемарљивом у односу на Ri и узима се да је 0. Такође, мотор производи много већи момент него што губи због треће осовине, те се узима да је и $b = 0$. Једначине (2.28) и (2.29) сада постају:

$$V_0 = iR + V_e \quad (2.30)$$

$$\tau = M_0 + I_M \dot{\omega} \quad (2.31)$$

Даље се могу извести једначине за M_0 :

$$i = \frac{V_0}{R} - \frac{k_e}{R} \omega \quad (2.32)$$

$$M_0 = k_i i - I_M \dot{\omega} = \frac{k_i}{R} V_0 - \frac{k_i k_e}{R} \omega - I_M \dot{\omega} \quad (2.33)$$

Мотори се налазе између тачкова и тела робота, што значи да постоји веза између θ , φ и ω :

$$\omega = \dot{\varphi} - \dot{\theta} \quad (2.34)$$

Убацавањем једначине (2.34) у једначину (2.33) добија се:

$$M_0 = \frac{k_i}{R} V_0 - \frac{k_i k_e}{R} (\dot{\varphi} - \dot{\theta}) - I_M (\ddot{\varphi} - \ddot{\theta}) \quad (2.35)$$

2.3 Динамички модел система

Из једначина (2.23) и (2.35) добија се:

$$\begin{bmatrix} 2I_t + 2I_M + 2m_t R^2 + mR^2 & mRL\cos\theta - 2I_M \\ mRL\cos\theta - 2I_M & I + 2I_M + mL^2 \end{bmatrix} \begin{bmatrix} \ddot{\varphi} \\ \ddot{\theta} \end{bmatrix} + \begin{bmatrix} \beta_{os} + \beta_{tr} + \frac{2k_i k_e}{R} & -\beta_{os} - \frac{2k_i k_e}{R} \\ -\beta_{os} - \frac{2k_i k_e}{R} & \beta_{os} + \frac{2k_i k_e}{R} \end{bmatrix} \begin{bmatrix} \dot{\varphi} \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & -mgL \end{bmatrix} \begin{bmatrix} \varphi \\ \theta \end{bmatrix} = \begin{bmatrix} \frac{2k_i}{R} \\ -\frac{2k_i}{R} \end{bmatrix} V_0 \quad (2.36)$$

Записано у краћем облику:

$$\mathbf{E} \begin{bmatrix} \ddot{\varphi} \\ \ddot{\theta} \end{bmatrix} + \mathbf{F} \begin{bmatrix} \dot{\varphi} \\ \dot{\theta} \end{bmatrix} + \mathbf{G} \begin{bmatrix} \varphi \\ \theta \end{bmatrix} = \mathbf{H} V_0 \quad (2.37)$$

Нека су матрице \mathbf{P} , \mathbf{Q} и \mathbf{R} дефинисане као:

$$\mathbf{P} = -\mathbf{E}^{-1}\mathbf{G}, \mathbf{Q} = -\mathbf{E}^{-1}\mathbf{F}, \mathbf{R} = \mathbf{E}^{-1}\mathbf{H} \quad (2.38)$$

Трансформацијом једначине (2.36) добија се једначина стања самобалансирајућег робота на два тачка:

$$\frac{d}{dt} \mathbf{x} = \begin{bmatrix} \dot{\varphi} \\ \dot{\theta} \\ \ddot{\varphi} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \mathbf{P}(1,1) & \mathbf{P}(1,2) & \mathbf{Q}(1,1) & \mathbf{Q}(1,2) \\ \mathbf{P}(2,1) & \mathbf{P}(2,2) & \mathbf{Q}(2,1) & \mathbf{Q}(2,2) \end{bmatrix} \begin{bmatrix} \varphi \\ \theta \\ \dot{\varphi} \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ \mathbf{R}(1,1) \\ \mathbf{R}(2,1) \end{bmatrix} V_0 = \mathbf{A} \mathbf{x} + \mathbf{B} V_0$$

(2.39)

$$\mathbf{y} = \begin{bmatrix} R & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \varphi \\ \theta \\ \dot{\varphi} \\ \dot{\theta} \end{bmatrix} = \mathbf{C}\mathbf{x} \quad (2.40)$$

где су $\varphi, \theta, \dot{\varphi}$ и $\dot{\theta}$ променљиве система. Улаз у систем је напон V_0 , а излази су угао отклона робота од вертикале θ и позиција робота $x = R\varphi$ на хоризонталној оси по којој се робот креће.

Контролабилност је веома важна особина контролних система и игра важну улогу у проблемима контроле као што су стабилизација нестабилних система помоћу повратне спреге или проналажење оптималне контроле. За систем се каже да је потпуно контролабилан уколико се за било које почетно и жељено стање може одредити управљачка секвенца, тако да систем после одређеног времена почев од почетног стања стигне у жељено стање. Из једначине стања се може одредити да ли је систем контролабилан или не, применом одређених математичких трансформација. Контролабилност система описаног са

$$\dot{\mathbf{x}} = \mathbf{A}\mathbf{x} + \mathbf{B}u \quad (2.41)$$

проверава се одређивањем ранга матрице \mathbf{K} , где је \mathbf{K} дефинисано са:

$$\mathbf{K} = [\mathbf{B} \quad \mathbf{A}\mathbf{B} \quad \mathbf{A}^2\mathbf{B} \quad \mathbf{A}^3\mathbf{B}] \quad (2.42)$$

Убацавањем свих параметара робота добијено је да је систем потпуно контролабилан за сваки улаз V_0 .

2.4 Сензор

Приликом конструисања робота први проблем је одређивање угла отклона робота од равнотежног положаја. Ово је могуће урадити на више начина: помоћу акцелерометра, жirosкопа или оба сензора заједно.

2.4.1 Акцелерометар

Акцелерометар је електромеханички уређај који се користи за мерење убрзања објекта на који је постављен. Убрзање објекта зависи од сила које делују на њега. Ове силе могу бити статичке, као што је константна гравитациона сила која делује на доле, или динамичке, проузроковане кретањем или вибрацијама акцелерометра. Одређивање угла под којим је робот нагнут може се добити једноставним тригонометријским трансформацијама вредности убрзања које мери акцелерометар. Проблем при одређивању угла на овај начин настаје јер све хоризонталне или вертикалне вибрације акцелерометар детектује као промену убрзања. Ова грешка се може уклонити применом Калмановог филтра или нискофреквентног филтра у имплементацији са комплементарним филтром. На овај начин пропуштају се само ниске фреквенције, док се све више одстрањују, те се кратке, тренутне промене убрзања не детектују. Применом ових филтара на добијени сигнал из акцелерометра смањује се утицај вибрација и добија се тачније мерење.

2.4.2 Жироскоп

Жироскоп је уређај који се користи за мерење угаоне брзине објекта на који је постављен. Израчунавање угла отклона објекта од равнотежног положаја врши се нумеричком интеграцијом угаоне брзине коју даје жироскоп. У равнотежном положају израчунати угао би увек требало да буде 0. Међутим, због нумеричке интеграције долази до акумулирања грешке, те се после неког времена у равнотежном положају добијају и ненулте вредности угла. Овај проблем се може делимично решити коришћењем високофреквентног филтра у имплементацији са комплементарним филтром или Калмановим филтром. На овај начин добијају се прецизнија мерења жироскопа.

У пракси се врло ретко користе само жироскоп или само акцелерометар за одређивање угла, већ се користе оба сензора, па се вредности оба мерења комбинују.

2.4.3 Моделирање сензора

Како микроелектромеханички (МЕМС) сензори нису идеални, за веродостојну симулацију потребно је моделирати сензоре са њиховим грешкама. Температурни ефекти се занемарују у симулацији јер систем није предвиђен да се креће по простору променљиве температуре.

Карактеристике грешке сензора:

1. **Стална грешка**

Стална грешка је грешка која је присутна све време и расте линеарно.

2. **Термо-механички шум**

Термо-механички шум се јавља као нуспојава термо-електричног шума фреквенције много веће од фреквенције одабирања сензора. Последица је појава шума у виду случајне шетње. (*random walk*)

3. **Трептајни шум**

Због треперења електроничких делова, трептајни шум се појављује у ниским фреквенцијама спектра измереног сигнала у виду белог шума.

2.5 Систем за управљање

2.5.1 Филтри

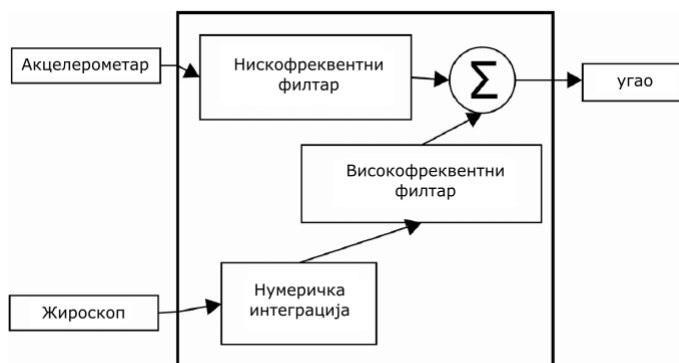
С обзиром на то да ни акцелерометар ни жirosкоп нису довољно прецизни мерни уређаји, користе се различити филтри који исправљају њихове недостатке. Најчешће коришћени су комплементарни филтар и Калманов филтар. Калманов филтар је теоријски идеалан за комбиновање вредности измерених помоћу сензора и користи се у професионалним системима за навигацију. Међутим, Калманов филтар је комплексан и због великог броја операција веома захтеван за процесор. Комплементарни филтар је мање процесорски захтеван и мање прецизан, али довољно прецизан за потребе управљања самобалансирајућег робота.

Комплементарни филтар

Улази комплементарног филтра су угаона брзина објекта измерена жirosкопом и угао под којим се објекат налази у односу на вертикалу добијен помоћу акцелерометра. Састоји се од нискофреквентног филтра, нумеричке интеграције и високофреквентног филтра, као што је и приказано на слици 2.3.

1. **Нискофреквентни филтар**

Нискофреквентни филтар је у овом раду коришћен за обраду података добијених са акцелерометра. Овај филтар уклања високе фреквенције из сигнала што су у овом случају фреквенције шума насталог вибрацијама. Ако се угао који читава акцелерометар нагло промени са нула на десет степени, излаз филтра неће одмах постати десет степени, већ ће се постепено повећавати и после неког времена доћи до те вредности. После



Слика 2.3: Приказ комплементарног филтра

употребе филтра остају само дуготрајне промене, изазване нагињањем робота. Недостатак је кашњење које настаје због постепене промене угла. Једначина нискофреквентног филтра:

$$\alpha(t) = (1 - k)\alpha(t - 1) + k\theta(t) \quad (2.43)$$

где је α угао који се добија на излазу филтра, θ угао који се добија са акцелерометра и доводи на улаз филтра, а k константна филтра. Тренутни угао акцелерометра добија се као:

$$\theta_{acc} = \arccos \frac{merenje_{acc}}{mg} \quad (2.44)$$

2. Нумеричка интеграција

Нумеричком интеграцијом добија се вредност угла отклона са жироскопа тако што се на тренутну вредност угла дода производ угаоне брзине и временске константе, која је у овом случају време између два узастопна одбирка сигнала са сензора. Једначина нумеричке интеграције у овом случају је:

$$\alpha(t) = \alpha(t - 1) + \omega(t)dt \quad (2.45)$$

где је α добијени угао, ω угаона брзина са жироскопа и dt период одабирања. Проблем акумулације грешке која неминовно настаје услед процеса интеграције минимизује се употребом високофреквентног филтра.

3. Високофреквентни филтар

Овај филтар ради потпуно супротно од нискофреквентног филтра и у овом случају има за циљ смањивање грешке жироскопа. Једначина високофреквентног филтра:

$$\alpha(t) = (1 - k)(\alpha(t - 1) + \omega(t)dt) \quad (2.46)$$

Сада се може извести једначина комплементарног филтра:

$$\alpha(t) = (1 - k)(\alpha(t - 1) + \omega(t)dt) + k\theta(t) \quad (2.47)$$

Период одабирања dt је величина дата у каталозима проидвојача сензора и обично корисник може изабрати између неколико вредности. Временска константа комплементарног филтра представља време током којег сигнал делује на филтар. Код нискофреквентног филтра, сигнали који трају доста дуже од временске константе пролазе неизмењени, док краћи сигнали буду елиминисани. Временска константа τ дата је једначином:

$$\tau = \frac{kdt}{1 - k} \quad (2.48)$$

где је k константа нискофреквентног филтра. На основу угла добијеног на излазу комплементарног филтра врши се управљање моторима.

Калманов филтар

За разлику од комплементарног филтра који користи само мерења са сензора, Калманов филтар користи и динамички систем за естимацију угла отклона.

Калманов филтар или линеарно квадратни естиматор има многобројне примене у индустрији. Користи се за навођење, навигацију и контролу возила. Главни разлог широке употребе овог филтра јесте његова способност да даје веома прецизна предвиђања уколико је динамички модел система веродостојан физичком систему.

Алгоритам филтра се састоји из два дела. У првом делу се на основу динамичког модела процењује тренутно стање на следећи начин:

$$\vec{x}_p = A\vec{x}_{n-1} + B\vec{u}_n \quad (2.49)$$

где је \vec{x}_p тренутно предвиђено стање, \vec{x}_{n-1} стање предвиђено у претходној итерацији алгоритма, а \vec{u}_n контролни вектор, односно сигнал актуатора. Матрица

A је матрица промене стања, а матрица B је контролна матрица која представља промене стања унете покретањем актуатора. Почетно стање система \vec{x}_0 представља очекивано стање система пре покретања.

У првом делу алгоритма процењује се и коваријанса предвиђања, односно мера грешке предвиђеног стања. Коваријанса предвиђања у датом тренутку се израчунава као:

$$P_p = AP_{n-1}A^T + Q \quad (2.50)$$

где је P_{n-1} коваријанса из претходне итерације, а Q је процењена грешка коваријансе процеса.

Други део алгоритма уводи мерења са сензора. Прво се рачуна иновација \tilde{y} која представља разлику предвиђеног стања и стања добијеног читавањима са сензора. Измерене вредности са сензора чине вектор \vec{z}_n , а матрица H пресликава вектор стања у вектор мерења.

$$\tilde{y} = z_n - Hx_p \quad (2.51)$$

Као и у првом делу, рачуна се коваријанса иновације S која представља стварну грешку у односу на предвиђање:

$$S = HP_pH^T + R \quad (2.52)$$

где је R процењена грешка коваријансе процеса.

Калманово појачање представља релативно појачање мерења у односу на предвиђено стање. Са високим појачањем филтер се више ослања на скорашња мерења, док ниско појачање даје предност динамичком моделу. Ако је појачање блиско јединици добијаће се веома нагле промене стања, а ако је појачање блиско нули стање ће бити инертно са малим шумом. Калманово појачање се рачуна као:

$$K = P_pH^T S^{-1} \quad (2.53)$$

Последњи корак алгоритма је крајња естимација тренутног стања:

$$\vec{x}_n = \vec{x}_p + K\tilde{y} \quad (2.54)$$

и рачуна се нова коваријанса предвиђања:

$$P_n = (I - KH)P_p \quad (2.55)$$

где је I матрица идентитета.

У овом раду коришћена су два сензора, жirosкоп и акцелерометар, па је вектор мерења:

$$\vec{z}_n = \begin{bmatrix} \theta_{accel} \\ \dot{\theta}_{accel} \\ \theta_{gyro} \\ \dot{\theta}_{gyro} \end{bmatrix} \quad (2.56)$$

2.5.2 Стабилизатори

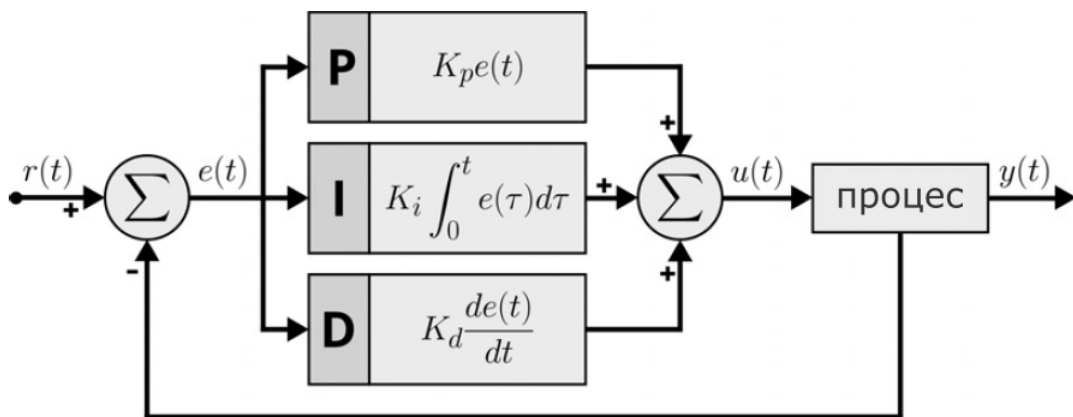
ПИД контролер

Када се робот изведе из равнотежног положаја робот ће пасти на страну на коју је отклоњен, осим ако се на њега не почне да делује сила у супротном смеру која би смањила угао отклона. Ова сила изазвана је радом мотора, односно окретањем точкова робота. Динамичким моделом система показано је како одређени напон на мотору утиче на угао отклона, позицију и брзину робота. Потребно је на неки начин у сваком тренутку израчунати колики напон треба предати моторима да би робот остао у равнотезном положају. Један од начина за то подразумева коришћење ПИД контролера.

ПИД контролер се састоји из три засебна контролера:

1. П - пропорционални део
2. И - интеграциони део
3. Д - деривациони (односно изводни) део

Сваки од ових контролера на улазу добија одступање тренутне вредности угла од референтне, односно жељене вредности, а на излазу даје напон који се шаље мотору. Блок шема ПИД контролера приказана је на слици 2.4.



Слика 2.4: Блок шема ПИД контролера

На слици 2.4 променљива r представља референтну вредност угла отклона, e разлику референтне и тренутне вредности, а u представља напон који ће бити предат моторима.

Једначина ПИД контролера је:

$$U = K_P e + K_I \int_0^t e(t) dt + K_D \frac{de}{dt} \quad (2.57)$$

Коефицијенти K_P , K_I И K_D се могу добити на више начина да би се добило жељено понашање контролера. У овом раду коришћен је *MATLAB PID Tuner* апликација за добијање ових параметара.

Линеарни квадратни регулатор

Теорија оптималне контроле се бави моделирањем контролера који испуњавају идеалне услове за стабилизацију динамичког модела са задатим улазним контролама. Динамички модел који је описан диференцијалним једначинама првог реда, а функција грешке квадратном функцијом назива се ЛК проблем. Једно од најпознатијих решења овог проблема јесте ЛКР, односно линеарни квадратни регулатор који затвара повратну спрегу по стањима. Подешавање регулатора који контролише неки систем врши се минимализацијом функције грешке. У овом случају грешка је отклон робота од равнотежног положаја. Предност ЛКР контролера јесте у томе што се подешавају само параметри, а не и сам контролер. Једначине динамичког система који контролер стабилизује дате су као:

$$\dot{x}(t) = Ax(t) + Bu(t) \quad (2.58)$$

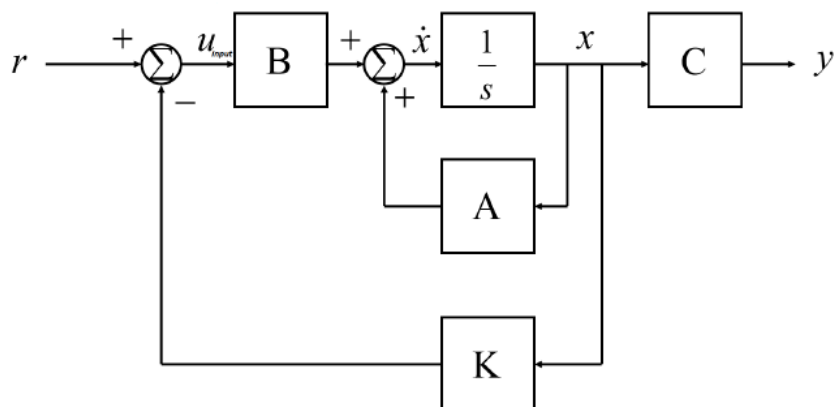
$$y(t) = Cx(t) \quad (2.59)$$

$$x(0) = x_0 \quad (2.60)$$

Функција грешке контролера је дата као:

$$J = \frac{1}{2} \int_0^{\infty} [x^T(t)Qx(t) + u^T(t)Ru(t)] dt \quad (2.61)$$

Блок шема ЛКР контролера дата је на слици 2.5.



Слика 2.5: Блок шема ЛКР контролера

Једначина по којој се добија u је:

$$u = -Kx \quad (2.62)$$

где је матрица K дефинисана са:

$$K = R^{-1}B^T P \quad (2.63)$$

Матрица P се добија из алгебарске Рикатијеве једначине:

$$Q + A^T P + P A - P B B^T P = 0 \quad (2.64)$$

Матрице Q и R балансирају утицај контролног улаза u и тренутног стања x у функцији грешке J . Матрица Q одређује колико брзо сваки од елемената вектора x долази до жељене крајње вредности. Она је облика:

$$Q = \begin{bmatrix} a & 0 & 0 & 0 \\ 0 & b & 0 & 0 \\ 0 & 0 & c & 0 \\ 0 & 0 & 0 & d \end{bmatrix} \quad (2.65)$$

где су a, b, c и d тежински коефицијенти за $\varphi, \theta, \dot{\varphi}$ и $\dot{\theta}$. Матрица R је у овом случају коефицијент, јер је контролни вектор u величине $[1, 1]$.

Главни циљ контролера је да доведе сва стања система, односно $\varphi, \theta, \dot{\varphi}$ и $\dot{\theta}$ до нуле у што краћем периоду. Међутим, што брже контролер доводи стања до жељене вредности, то је већа вредност улаза, односно напона у овом случају. Како напон на моторима не може бити бесконачан, праве се одређени компромиси при дизајнирању контролера и бирању његових параметара.

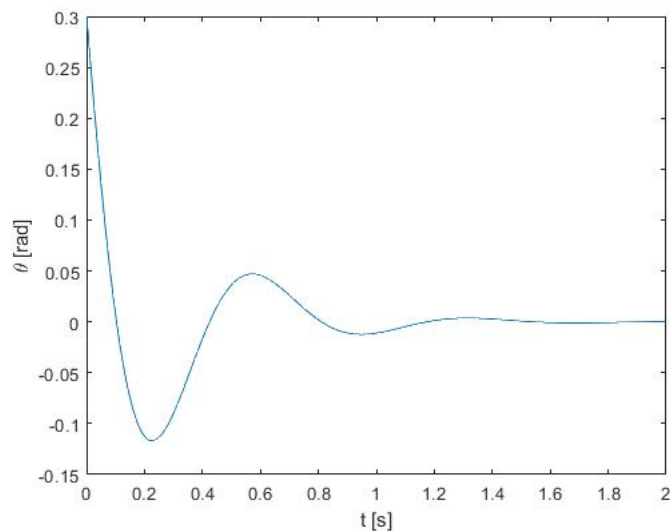
3

Симулација

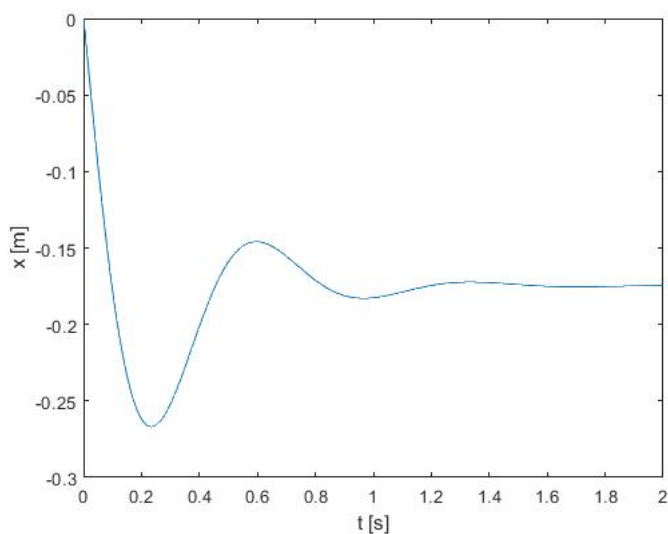
Динамички модел и контролер изведени у поглављу 2 имплементирани су у програму *MATLAB Simulink*. Параметри динамичког модела преузети су из рада [2]. Имплементирани су и ПИД и ЛКР контролер. Коефицијенти ПИД контролера добијени су помоћу *PID Tuner*-а и то су:

$$K_P = 0.03923, K_I = 0.1299, K_D = 0.0029 \quad (3.1)$$

На сликама 3.1 и 3.2 приказана је промена угла отклона и позиције робота приликом стабилизације робота помоћу ПИД-а. Почетни угао отклона је 0.3 радијана и нема додатних спољашних импулса.



Слика 3.1: График зависности угла отклона од времена



Слика 3.2: График зависности позиције од времена

Примећује се да ПИД стабилизује угао отклона за око од 1.5 секунди, али се пре стабилизације појављује неколико прескока до 0.1 радијана. На слици 3.2 види се да стабилизација позиције робота није успела у потпуности. Док се стабилизује угао, нагињање робота изазива кретање лево или десно што за резултат има то да се робот не враћа у почетну позицију већ остаје на 17 центиметара од места са ког је пошао. Разлог томе је што је ПИД контролер способан да стабилизује једну, али не и више стања система. Слично као и код угла отклона, и позиција има неколико прескока док не дође до константне вредности.

При ЛКР контролеру коришћене су следеће вредности параметара:

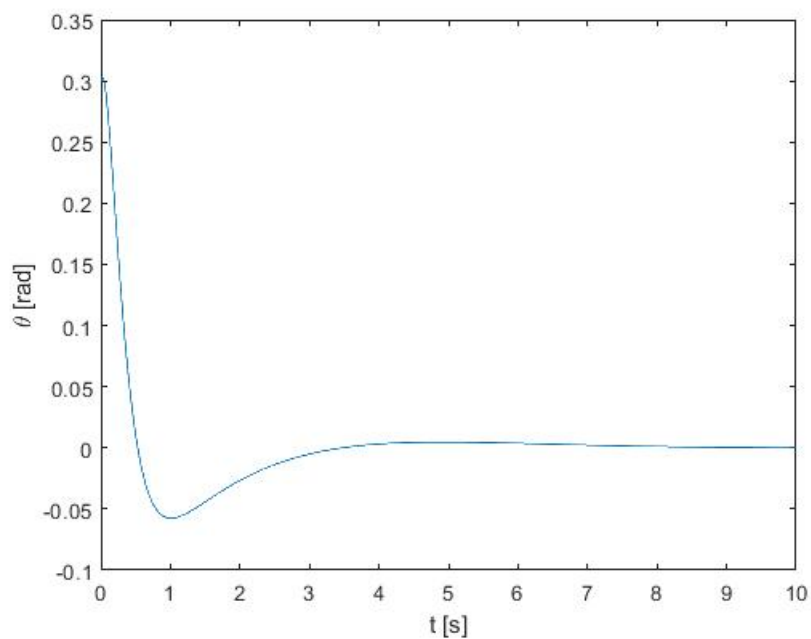
$$Q = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3.2)$$

$$R = 1 \quad (3.3)$$

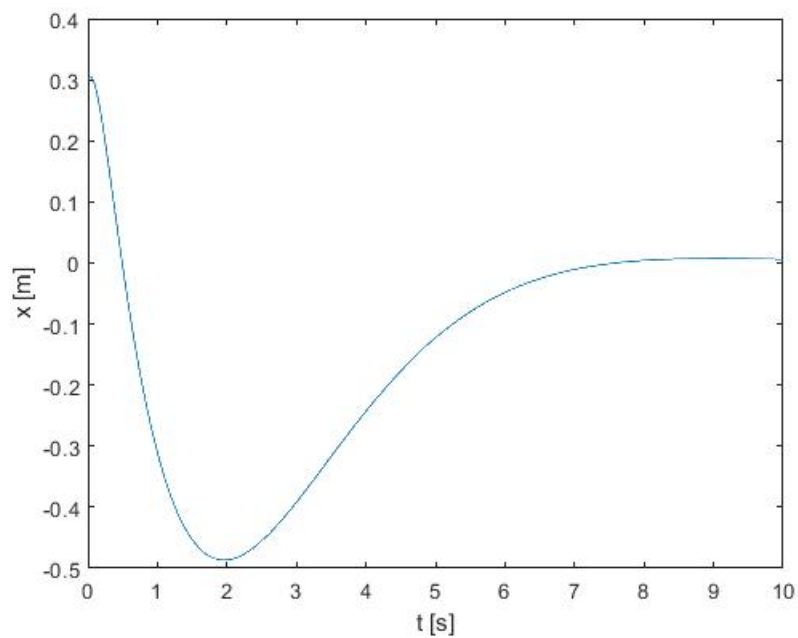
Помоћу Q и R у Matlab-у је добијена матрица K :

$$K = [-1.0000 \quad -2.5414 \quad 44.3545 \quad 8.3555] \quad (3.4)$$

На сликама 3.3 и 3.4 приказана је промена угла отклона и позиције робота приликом стабилизације робота помоћу ЛКР-а. Почетни угао отклона је, као и на сликама 3.1 и 3.2, 0.3 радијана и нема додатних спољашних импулса.



Слика 3.3: График зависимости угла отклонения от времени



Слика 3.4: График зависимости позиции от времени

За разлику од ПИД контролера, ЛКР успева да стабилизује оба стања система и доводи их до нуле. Угао отклона је стабилизован за око 3 секунде. Ово је више него код ПИД-а, али у овом случају постоји само један прескок и то мањи него у првом случају. Позиција јесте доведена до нуле, али тек након 7 секунди и са много већим прескоком него код ПИД контролера.

ПИД контролер брже стабилизује систем, али су промене координата много хаотичније него код ЛКР контролера. У физичком систему где је најчешће потребно да робот дође до одређене позиције, потребна је стабилизација и угла отклона и позиције. У том случају боље је изабрати ЛКР контролер. У неким случајевима је неопходан брзи одзив и брза стабилизација угла отклона, где је боље одабрати ПИД контролер.

4

Закључак

У овом раду описан је самобалансирајући робот и његов контролни систем. Изведен је математички модел тела робота и његових актуатора, односно ДЦ мотора. Упоредене су карактеристике различитих сензора и описани су филтри који се користе за обраду сигнала из тих сензора. Такође, описана су два контролера чије су перформансе упоређене у симулацији. Симулација је радјена у програмском пакету *MATLAB Simulink* на основу претходно изведеног модела. Показано је да оба контролера имају своје предности и мане и да је, у зависности од потреба робота који се дизајнира, могуће користити и један и други.

Захвалност. Овим путем бих желела да се захвалим свом ментору Јовици Миљисављевићу на помоћи не само при изради овог рада, већ и на подршци при ваннаставним пројектима у протеклих неколико година. Посебну захвалност дугујем професорки Наташи Чалуковић, која је у највећој мери заслужна за све што сам за време школовања у Математичкој гимназији научила и остварила.

Литература

- [1] B. Faldu, R. Savaliya, D. Ginoya, T. R. Shah. *Stable Control System of Two Wheel Self Balancing Robot*
- [2] P. Frankovsky, L. Dominik, A. Gmitterko, I. Virgala. *Modeling of Two-Wheeled Self-Balancing Robot Driven by DC Gearmotors*
- [3] Y. Ding, J. Gafford, M. Kunio. *Modeling, Simulation and Fabrication of a Balancing Robot*
- [4] M. A. A. Shuvom. *Modeling, implementation, simulation and comparison of different control theories on a two wheel self balancing robot in simulink*
- [5] T. Braunl and J. Pan. *Balancing a Two-Wheeled Autonomous Robot*