

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

GRGO ČUPIĆ

**LOPTICA NA PLATFORMI: SUSTAV ZA
ISPITIVANJE NAPREDNIH ALGORITAMA
UPRAVLJANJA**

Zagreb, 2011

Ovaj rad izrađen je na Zavodu za automatiku i računalno inženjerstvo, u Laboratoriju za robotiku i inteligentne sustave upravljanja (LARICS), pod vodstvom prof. dr. sc. Stjepana Bogdana i predan je na natječaj za dodjelu Rektorove nagrade u akademskoj godini 2010./2011.

Sadržaj:

1.	Uvod.....	1
1.1.	Postojeća istraživanja na području problema loptice na platformi	2
2.	Opći i specifični ciljevi rada	4
3.	Materijali i metode.....	5
3.1.	Matematički modeli sustava	6
3.1.1.	Matematički model loptice	6
3.1.2.	Matematički model platforme	9
3.1.3.	Matematički model neravnina na platformi.....	13
3.1.4.	Provjera ispravnosti matematičkih modela.....	15
3.2.	Sinteza algoritma upravljanja	18
3.2.1.	Upravljanje P regulatorom	18
3.2.1.1.	Simulacija kružnice	18
3.2.1.2.	Simulacija kvadrata	19
3.2.1.3.	Simulacija pravca	20
3.2.2.	Pomoćna upravljačka petlja za linearizaciju platforme.....	21
3.2.2.1.	Simulacija kružnice	22
3.3.	Projektiranje regulatora metodom Truxal-Guillemin	26
3.3.1.	Linearizacija matematičkog modela sustava.....	27
3.3.2.	Sinteza regulatora metodom Truxal-Guillemin.....	29
3.3.3.	Upravljanje projektiranim regulatorom.....	30
3.3.3.1.	Simulacija kružnice	31
3.3.3.2.	Simulacija kvadrata	32
3.3.3.3.	Simulacija pravca	33
3.4.	Realizirani sustav.....	35
3.4.1.	Platforma i loptica	36
3.4.2.	Sklopovlje	37
3.5.	Programska podrška	40
3.5.1.	Grafičko korisničko sučelje	40
3.5.1.1.	Upravljački dio.....	41
3.5.1.2.	Serijska veza.....	43

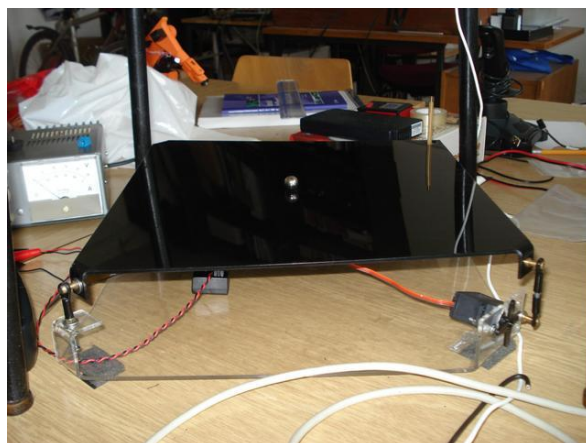
3.5.1.3.	Upravljanje igraćom palicom	43
3.5.1.4.	Zadavanje referentnog položaja loptice	44
3.5.1.5.	Iscrtavanje odziva	45
3.5.2.	Algoritam obrade slike	46
3.5.3.	Programiranje korisničkog algoritma upravljanja	49
4.	Rezultati i rasprava	51
4.1.	Eksperiment stabilizacije loptice u zadanoj točki.....	51
4.2.	Eksperiment kružnice	53
4.3.	Eksperiment kružnice bez pomoćne upravljačke petlje.....	55
4.4.	Eksperiment kvadrata	57
4.5.	Eksperiment kvadrata bez pomoćne upravljačke petlje	59
4.6.	Eksperiment pravca	61
4.7.	Eksperiment pravca bez pomoćne upravljačke petlje	63
4.8.	Rasprava	65
5.	Zaključci	66
6.	Zahvala	67
7.	Literatura	68
8.	Sažetak	70
9.	Summary.....	71
10.	Kratki životopis	72
Prilog A	Simulacijski odzivi - P regulator bez pomoćne upravljačke petlje	A-1
Prilog B	Simulacijski odzivi - P regulator sa pomoćnom upravljačkom petljom	B-1
Prilog C	Simulacijski odzivi - projektirani regulator sa pomoćnom upravljačkom petljom.....	C-1
Prilog D	Eksperimentalno korišteni pokazni algoritam upravljanja	D-1

1. Uvod

Predmet ovog rada je izrada edukacijskog modela i pokaznog upravljačkog algoritma za sustav loptice na nelinearnoj platformi izrađenoj u LARICS laboratoriju. Loptica koja se pozicionira na platformi primjer je inherentno nestabilnog i nelinearnog sustava. Sustav loptice na platformi (*eng.* ball-plate system) u žarištu je istraživanja od prve pojave edukacijskih uređaja dostupnih na tržištu od kraja 1980ih godina. Sustav loptice na platformi sa navedenim karakteristikama odgovora istraživanju naprednih algoritama upravljanja te ocjenjivanju njihovih prednosti i mana. Zbog navedenog mnoga su istraživanja na tom području provedena upravo na općeprihvaćenom problemu upravljanja sustava loptice na platformi.

Ovakvi sustavi mogu se susresti u energetici, kemijskoj, svemirskoj i drugim industrijama. Platforme koje služe za pozicioniranje objekata susreću se u avionskoj industriji kao simulatori leta, za održavanje vozila u automobilskoj industriji, za pozicioniranje satelita, za razne primjene u mostogradnji. Vjeran primjer je Stewart platforma koja ima 6 stupnjeva slobode. Stewartova platforma se tako koristi u sklopu NASA-inog LIDS-a (Low Impact Docking System) koji služi za upravljanje pristajanjem svemirskih letjelica u svemirske postaje.

Fizički sustav korišten u ovom radu sastoji se od platforme s dva stupnja slobode, pokretane motornim servomehanizmom. Na platformi se nalazi loptica koju je moguće pomicati u dva smjera manipulacijom platforme (slika 1.1). Sustav loptice na platformi inherentno nije linearan, no dodatni problem pri upravljanju stvara sama ploha platforme koja nije u potpunosti ravna što unosi dodatne nelinearnosti.



Slika 1.1 Korištena platforma s lopticom

Ovaj rad obuhvaća postupke analize i sinteze ovakvog sustava automatskog upravljanja, simulacije provedene u Matlab Simulink okruženju te izradu programske infrastrukture na osobnom računalu koja, osim što implementira upravljanje na realnom sustavu, omogućava korisniku izradu i testiranje vlastitih algoritama upravljanja bez upotrebe drugih programskih razvojnih alata. Povratna veza realizirana je web-kamerom s koje se obrađuje slika te dobiva točna pozicija loptice na platformi.

Kratka video prezentacija realiziranog sustava može se pronaći na sljedećoj Internet adresi: <http://www.youtube.com/watch?v=e1nrvpwmznY> ili youtube pretragom: „Ball and Plate: Advanced Controller Design Testing System“

1.1. Postojeća istraživanja na području problema loptice na platformi

Sustav loptice na platformi postao je popularni predmet istraživanja mnogih znanstvenika na polju automatskog upravljanja krajem 1980ih pojavom edukativnih modela na tržištima diljem svijeta. Mnogi znanstveni članci bave se problemom upravljanja loptice na platformi. Pristupi problemu, kako i korišteni algoritmi, razlikuju se od autora do autora. Ipak mogu se naći slična rješenja povratne veze, većina autora koji su implementirali algoritme na stvarnom sustavu kao povratnu vezu koristili su sliku s kamere. Rijetko koji autori odlučili su se za vlastitu izradu platforme, većina ih koristi „HUMUSOFT CE151 Ball&Plate Apparatus“ edukacijski model ili BPVS-JLU-II razvijen pri Fakultetu komunikacijskog inženjerstva Sveučilišta Jilin u Kini. Kao upravljački algoritmi koriste se klasične strukture regulatora kao i moderni napredni algoritmi upravljanja.

U nastavku su navedeni različiti objavljeni radovi s temom upravljanja loptice na platformi te specifičnosti kod izvedenog rješenja:

- Autori u [1] koriste klasični PID/PSD regulatora te Humusoftov CE151 edukacijski model
- Autori u [2] realizirali su povratnu vezu korištenjem ekrana osjetljivog na dodir te klasičnog PID regulatora u HID strukturi. Pokazano je da je ekran osjetljiv na dodir gotovo najbolji izbor za povratnu vezu zbog svojih karakteristika koje garantiraju veliku točnost i preciznost. Ipak ovakvo rješenje iziskuje velike financijske izdatke.
- Autori u [3] projektirali su neizraziti *fuzzy* regulator realiziran neuronskom mrežom te optimiziran genetskim algoritmom (*GA-FNNC*).
- Autori u [4] predlažu kaskadni sustav upravljanja s aktivnim regulatorom poremećajne veličine (ADRC) dok u [5] nelinearni kaskadni regulator. Korišten je kineski BPVS-JLU-II.
- Autori u [6] koriste supervizorski neizraziti *fuzzy* regulator sa dva sloja upravljanja, donji sloj sastoji se od SIRM *fuzzy* pravila a gornji sloj provodi nadzor nad brzinom i akceleracijom loptice. Korišten je kineski BPVS-JLU-II.
- Autori u [7] predlažu regulator projektiran po ISE optimizacijskom kriteriju. Koristi se Humusoft CE151 edukacijski model.
- Autori u [8] autori izradili su nelinearni obzerver brzine na kineskom BPVS-JLU-II.
- Autori u [9] predlažu PID kontroler sa *fuzzy* supervizorom koristi se vizualna povratna veza te platforma vlastite izrade.

Iz navedenog se vidi da rješenje problema nije jednoznačno ali da različiti pristupi rješavanju problema dovode do jednako kvalitetnih rezultata. Navedeni radovi nisu razmotrili realizaciju LQR ili MPC upravljačkih algoritama koji bi svakako bili adekvatni za ovakav sustav.

Problem upravljanja lopticom na platformi bio je predmet istraživanja u Laboratoriju za robotiku i inteligentne sustave upravljanja (LARICS). Sva istraživanja izvedena su korištenjem LARICS platforme s dva stupnja slobode koja je izrađena za ovu namjenu.

Upravljanje P regulatorima u kaskadnoj strukturi upravljanja izvedeno je u [10]. Kao programski alat koristio se Matlab Simulink a povratna veza realizirana je jednostavnom web-kamerom. Autor u [11] provodi dekompoziciju procesa te upravljanje viševarijabilnim regulatorom, programski alat također je Matlab Simulink te povratna veza web-kamerom. Autor u [12] koristi neizraziti *fuzzy* regulator realiziran na mikroprocesorskom *Gumstix* sustavu uz vizualnu povratnu vezu s računala. Autor u [13] realizira i vizualnu povratnu vezu i upravljački algoritam na mikroprocesorskom *Gumstix* sustavu.

Ovaj rad nastavak je istraživanja na području problema loptica na platformi te ima višestruke konačne ciljeve. Prvi i osnovni cilj je izrada potrebne programsko sklopovske infrastrukture kako bi korištena platforma postala edukacijski model za istraživanje i ispitivanje, a drugi je izrada odgovarajućeg upravljačkog algoritma.

2. Opći i specifični ciljevi rada

Korišteni sustav loptice s platformom sastoji se od kuglice te nelinearne platforme s dva stupnja slobode koju pokreću dva servomotora. Platforma nije ravna ploha već izvitoperena u smjeru slobodnog vrha. Zakrivljenost platforme predstavlja dodatan problem pri upravljanju pošto kretanje loptice direktno ovisi i o zakrivljenosti. Sustav loptice na platformi inherentno je nelinearan, nestabilan i dinamičan. Cilj ovog rada je projektirati odgovarajući pokazni algoritam upravljanja koji će ovakav sustav prvenstveno stabilizirati te zatim upravljati pozicijom loptice na platformi s ciljem izvođenja različitih tipova trajektorija.

Važniji cilj, koji proizlazi iz potrebe upravljanja ovakvim sustavom, realizacija je odgovarajuće programske i sklopovske infrastrukture koja će omogućiti upravljanje i ispitivanje novih tipova algoritama na realnom sustavu. Sukladno ciljevima u radu izrađena je Windows aplikacija koja realizira osnovne funkcionalnosti kao što su: hvatanje pozicije loptice na platformi preko slike dobivene s web-kamere, upravljanje radom servomotora, ručno upravljanje pomoću palice za igru (*eng. Joystick*), izvođenje projektiranog upravljačkog algoritma, zadavanje reference tj. trajektorije gibanja kuglice, iscrtavanje odziva sustava (pozicije loptice, upravljačkih signala) te najvažnija funkcija omogućavanje unošenja i pokretanja novog algoritma upravljanja koji korisnik može definirati i oblikovati po vlastitim željama tj. potpuno je slobodan u stvaranje funkcije upravljanja.

Ovako izrađen sustav daje potpuno neovisnu, samostalnu podlogu da se korišteni sustav loptice na platformi koristi kao edukacijski model koji može služiti za učenje ili ispitivanje kako klasičnih tako i naprednih algoritama upravljanja.

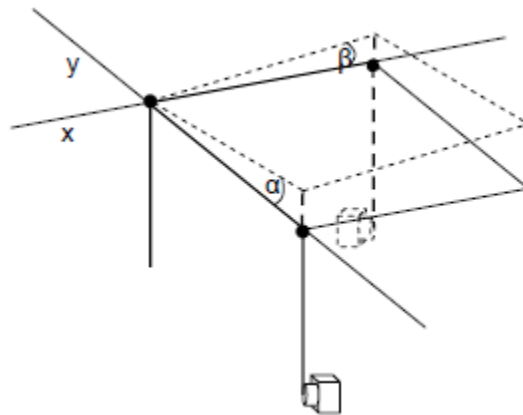
3. Materijali i metode

U nastavku su pobliže objašnjene metode i realizirana rješenja u različitim programskim alatima. Rješavanju problema upravljanja ovakvog sustava pristupa se sličnim načelnim postupkom kao i s bilo kojim sustavom automatskog upravljanja. Provodi se analiza, sinteza, simulacije sustava te naposljetku realizacija na realnom sustavu.

Osnova rješavanja problema upravljanja je izrada matematičkog modela loptice i platforme (analiza). Jednadžbe koje matematički opisuju ponašanje sustava realizirane su u Matlab Simulinku pomoću skupa blokova koji će ih opisivati te ovisno o ulaznim varijablama proračunavati upravljački signal te poziciju loptice na platformi. Poznavajući matematički model sustava pristupa se postupku sinteze. Projektiraju se različiti algoritmi upravljanja sustava od kojih je konačno izabran onaj koji najviše udovoljava kriterijima kvalitete upravljanja sustava. Postupak izbora odgovarajućeg algoritma upravljanja provodi se simulacijama u Matlab Simulink okruženju. Nakon svega slijedi realizacija algoritma upravljanja na realnom sustavu. Provode se testiranja realnog ponašanja sustava te se donosi konačna ocjena o kvaliteti upravljanja.

3.1. Matematički modeli sustava

Sustav razmatran u ovom radu sastoji se od platforme s dva motorna mehanizma od kojih svaki zakreće svoj kut platforme (slika 3.1). Platforma je učvršćena u tri točke, svaka na jednom vrhu platforme. Četvrti vrh nije učvršćen ali postoji vodilica koja prilikom gibanja platforme osigurava pomicanje četvrtog ugla platforme samo po okomitoj osi. Okretanjem motora platforma se pomiče oko jedne i druge osi (dva stupnja slobode). Zadatak sustava upravljanja je odrediti kut za koji se platforma mora pomaknuti kako bi se loptica dovela u željeni položaj ili kako bi loptica pratila zadanu trajektoriju.



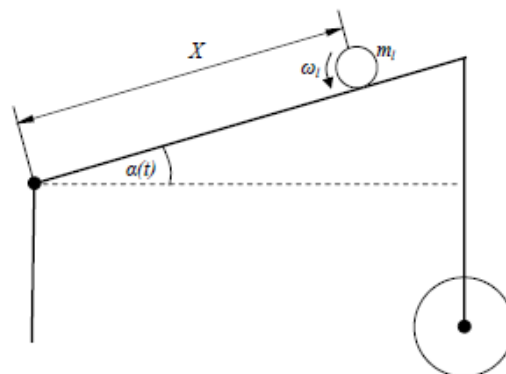
Slika 3.1: Platforma s motorima

Matematički modeli sustava izvedeni su u podsustavima: matematički model platforme, matematički model nelinearnosti na platformi te na matematički model loptice.

3.1.1. Matematički model loptice

Gibanje loptice po platformi opisano je matematičkim jednažbama. Prilikom konstrukcije modela u obzir se uzimaju sile koje djeluju na lopticu i uzrokuju njezino gibanje po platformi.

Translacijsko gibanje loptice po platformi uzrokovano je nagibom platforme i gravitacijskom silom koja djeluje na lopticu. Loptica se tada može pokazati kao tijelo na kosini koje počinje rotacijsko gibanje u smjeru pada kosine. Shematski prikaz je dan na slici 3.2.



Slika 3.2: Shematski prikaz loptice na platformi

Kut $\alpha(t)$ označava nagib platforme u odnosu na vodoravnu os. Vrijednost ω_1 predstavlja kutnu brzinu loptice dok vrijednost m_l predstavlja masu loptice koja iznosi. Pomak odnosno pozicija loptice po x osi na platformi označena je duljinom X . Na lopticu djeluju dvije sile; sila u smjeru gibanja $mg\sin\alpha$ te sila trenja koja djeluje suprotno smjeru gibanja loptice na platformi, označava se vektorom f . Ekvivalentno vrijedi i za parametre u smjeru y osi jednom razlikom. U smjeru y osi na platformi kut platforme u odnosu na vodoravnu os je označen sa $\beta(t)$.

Translacijska sila koja djeluje na lopticu u smjeru x osi definirana je sa:

$$F_{tx} = m_l g \sin\alpha \quad (3-1)$$

gdje je m_l masa loptice iznosa $m_l = 0.03 \text{ kg}$, a g je gravitacijska konstanta iznosa $g = 9.81 \text{ m/s}^2$. Trenje između loptice i platforme uzrokuje rotiranje loptice. Kada ne bi postojalo trenje između loptice i platforme, loptica bi klizila po platformi bez rotiranja oko vlastite osi. Ovaj moment je, dakle, izazvan silom trenja koja je suprotnog smjera od smjera gibanja loptice po platformi. Kako u nekom trenutku ipak dođe do gibanja loptice po platformi, znači da postoji moment koji se opire sili trenja. U ovom slučaju to je moment inercije loptice. Korištenjem drugog Newtonovog zakona za objekt u gibanju dobije se slijedeći izraz za navedeni slučaj:

$$M_r = R F_{tr} = J \cdot a = J \frac{\ddot{x}}{R} \quad (3-2)$$

gdje je: R radijus loptice iznosa $R = 0.0115 \text{ m}$, F_{tr} – sila trenja, a – akceleracija loptice, J – moment inercije loptice. Moment inercije loptice je određen izrazom:

$$J = \frac{2}{5} m_l R^2 \quad (3-3)$$

Uvrštavanjem (3-2) u (3-3) dobije se:

$$F_{rx} = \frac{2}{5} m_l \ddot{x} \quad (3-4)$$

Ovime su opisane sve sile koje djeluju na lopticu koja se giba po platformi i koje opisuju to gibanje slijedećom jednadžbom:

$$m_l \ddot{x} = \sum F = F_{tx} - F_{rx} = m_l g \sin\alpha - \frac{2}{5} m_l \ddot{x} \quad (3-5)$$

Nakon preuređivanja jednadžbe (3-5) dobije se jednostavniji izraz:

$$\ddot{x} = \frac{5}{7} g \sin\alpha \quad (3-6)$$

koji je vrlo jednostavno prebaciti u Laplaceovo područje kako bi se dobila prijenosna funkcija koja opisuje ovisnost promjene položaja loptice x ovisno o nagibu platforme α u smjeru x -osi. Nagib

platforme izražen je u radijanima i relativno je malog iznosa tako da vrijedi $\sin \alpha \approx \alpha$ što rezultira prijenosnom funkcijom:

$$\frac{X(s)}{\alpha(s)} = \frac{5 g}{7 s^2} \quad (3-7)$$

Izvod jednadžbi (3-1) do (3-7) vrijedi i za gibanje loptice po platformi u smjeru y osi. Jedine razlike su u označavanju ulaza i izlaza, odnosno nagiba platforme i pomaka loptice na platformi. Nagib platforme u smjeru y -osi označen je kutem β , dok je izlaz položaj loptice na y osi Y .

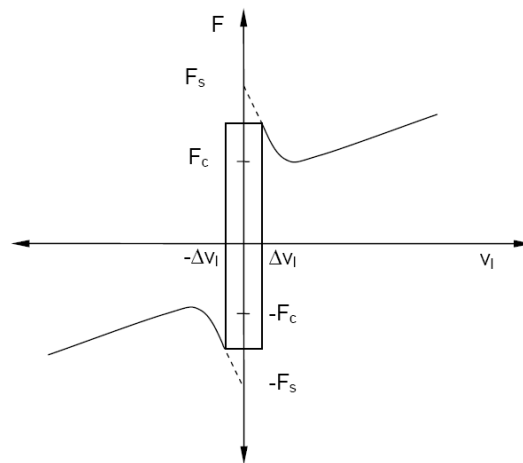
U model gibanja loptice potrebno je uvrstiti i model trenja. Trenje je modelirano i opisano klasičnim statičkim modelom trenja, čiji je oblik dan Slikom 3.3.

Model uključuje Coulombovo trenje:

$$F_c = \mu F_N \quad (3-8)$$

i viskozno trenje:

$$F_{fv}(v) = d \cdot v \quad (3-9)$$



Slika 3.3: Statički model trenja

Minimalna sila koju loptica mora savladati da bi počela svoje gibanje po platformi, neovisno o smjeru gibanja, određena je statičkim trenjem F_s .

Matematički izraz za cjelokupni model trenja dan je izrazom:

$$F_{tr} = \left| F_c + (F_s - F_c) e^{-\left(\frac{v}{v_s}\right)^\delta} \right| \operatorname{sgn}(v) \quad (3-10)$$

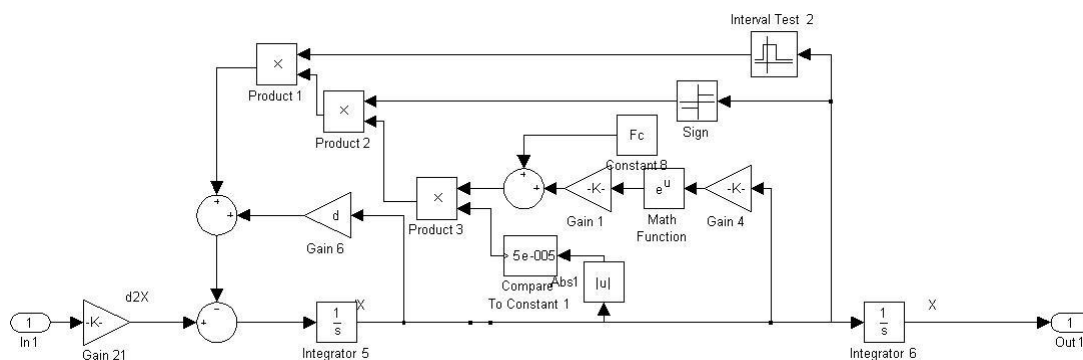
Gdje je:

- $F_c = 1.9935 \cdot 10^{-7}$ - Coulombovo trenje,

- $F_s = 9.9675 \cdot 10^{-8}$ - statičko trenje
- $v_s = 0.05 \text{ m/s}$ -Stribeckova brzina vrtnje,
- $\delta = 1$ -koeficijent
- $d = 2$ -koeficijent viskozno trenja

Stribeckova brzina vrtnje označava područje brzine gibanja loptice po platformi za koje vrijedi Stribeckov efekt koji dovodi do nelinearne promjene, odnosno nekontinuiranog pada sa statičkog trenja na Coulombovo trenje.

Model loptice i trenja izrađen u Simulinku za jednu od osi prikazan je na slici 3.4. Podsustav kao ulaz prima nagib platforme a kao izlaz daje trenutnu poziciju loptice na platformi.



Slika 3.4: Model loptice i trenja izrađen u Simulinku

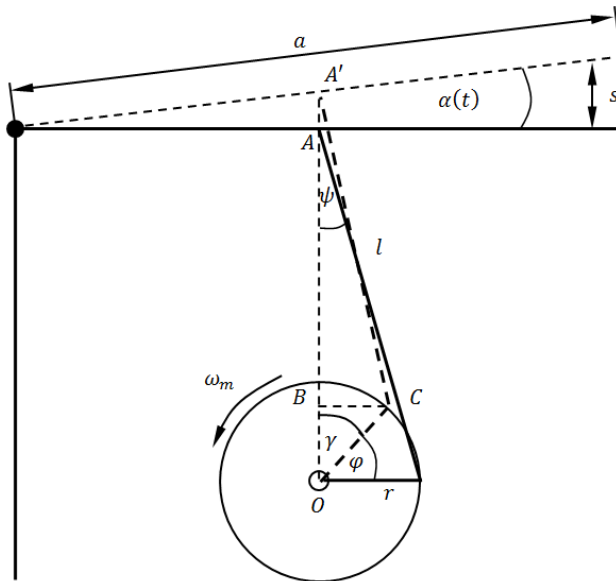
3.1.2. Matematički model platforme

Osnovni zadatak matematičkog modela platforme je da, ovisno o zakretu osovine aktuatorskih servomotora proračunava nagib platforme u odnosu na vodoravnu os. U stvarnosti platforma nije savršeno ravna ploha već je neravna te je zbog toga nagib platforme različit u svakoj točki platforme. Iz tog razloga matematički model platforme razložen je na dva podsustava.

Prvi podsustav matematičkog modela platforme opisuje koji je nagib platforme za slučaj kada bi platforma bila savršeno ravna i kruta ploha. Prvi podsustav u obzir uzima samo zakret osovine aktuatorskih servomotora. Drugi podsustav uzima u obzir trenutni položaj loptice na platformi te na osnovu njega proračunava odgovarajući nagib platforme u smjeru x i y osi, uzimajući u obzir zakrivljenost platforme. Izlazni proračun iz drugog podsustava predstavlja stvarnu informaciju o nagibu platforme.

U nastavku je dan izvod ovisnosti nagiba savršeno ravne krute platforme u odnosu na vodoravnu os u ovisnosti o zakretu osovine aktuatorskih servomotora. Pri izvodu se uvode određena pojednostavljenja. Pretpostavljeno je da zakret jednog servomotora uzrokuje podizanje samo jedne stranice platforme tj. da s pojedinim servomotorom se upravlja gibanjem loptice po jednoj osi. Zbog nesavršenosti mehaničke konstrukcije platforme ovo ne odgovara stvarnosti. U stvarnosti sustav je spregnut te se zakretom pojedinog servomotora utječe na gibanje loptice po obje osi.

Na Slici 3.5. shematski su prikazane i naznačene sve relevantne fizikalne veličine za ovaj proračun.



Slika 3.5: Zakret servomotora i platforme

Fizikalne veličine na slici 3.5. su redom:

- $r = 0.02 \text{ m}$ - polumjer motornog mehanizma,
- $l = 0.1 \text{ m}$ - duljina koljena motornog mehanizma,
- $\varphi(t) \text{ rad}$ - kut zakreta motora,
- $\omega_m(t) \text{ rad/s}$ - kutna brzina motora,
- $\alpha(t), \beta(t) \text{ rad}$ - kutevi zakreta platforme oko x-osi, odnosno y-osi,
- $s(t) \text{ m}$ - put koji platforma prijeđe pri zakretu za neki kut,
- $a = 0.3 \text{ m}$ - duljina i širina platforme.

Za određivanje kinematike i dinamike motora i platforme potrebno je promatrati rotaciju platforme za kut $\alpha(t)$ oko x odnosno istovjetno za kut $\beta(t)$ oko y osi.

Gibanje mehanizma se promatra od trenutka kada je platforma u srednjem položaju (ravna), odnosno kada je kut $\varphi(t) = 0$. Tada je put $s(t)$ koji platforma prevali pri pomicanju jednak nuli. Dužina \overline{AO} može se izraziti jednadžbom:

$$\overline{AO} = \sqrt{l^2 - r^2} \quad (3-11)$$

U vremenu dt motor se okrene za kut $\varphi(t) = \frac{d\varphi}{dt}$, a vrh koljena motora se pozicionira u točku A' . Iz Slike 3.5. jasno je da je platforma prevalila put:

$$ds = \overline{A'O} - \overline{AO} \quad (3-12)$$

U svrhu lakšeg modeliranja platforme uvode se dodatne točke B i C , te se dobivaju dva pravokutna trokuta $\Delta A'BC$ i ΔBCO . Prevaljeni put se može izraziti jednadžbom:

$$s(t) = \overline{A'B} + \overline{BO} - \overline{AO} \quad (3-13)$$

Budući da vrijedi:

$$\begin{aligned} \overline{A'B} &= l \cos \psi(t), \\ \overline{BO} &= r \cos \gamma(t) \end{aligned} \quad (3-14)$$

Jednadžba (3-13) može se pisati kao:

$$s(t) = l \cos \psi(t) + r \cos \gamma(t) - \sqrt{l^2 - r^2} \quad (3-15)$$

Budući da je potrebno izraziti ovisnost puta koji prevažuje platforma $s(t)$ i kuta motora $\varphi(t)$, jednadžba (3-15) postaje:

$$s(t) = l \cos \psi(t) + r \cos \left[\frac{\pi}{2} - \varphi(t) \right] - \sqrt{l^2 - r^2} \quad (3-16)$$

Drugi član ove jednadžbe može se raspisati kao:

$$\cos \left[\frac{\pi}{2} - \varphi(t) \right] = \cos \frac{\pi}{2} \cos \varphi(t) + \sin \frac{\pi}{2} \sin \varphi(t) = \sin \varphi(t) \quad (3-17)$$

Kako bi se eliminirao kut $\psi(t)$, odsječak \overline{BC} se izražava kao:

$$\overline{BC} = l \sin \psi(t) = r \sin \gamma(t) = r \sin \left[\frac{\pi}{2} - \varphi(t) \right] = r \cos \varphi(t) \quad (3-18)$$

Iz čega slijedi:

$$\sin \psi(t) = \frac{r \cos \varphi(t)}{l} \quad (3-19)$$

Iz jednadžbe (3-19) nadalje slijedi:

$$\cos \psi(t) = \sqrt{1 - \sin^2 \psi(t)} = \sqrt{1 - \left(\frac{r}{l} \right)^2 \cos^2 \varphi(t)} \quad (3-20)$$

Uvrštavanjem jednadžbi (3-17) i (3-20) u jednadžbu (3-16) dobiva se:

$$s(t) = l \sqrt{1 - \left(\frac{r}{l} \right)^2 \cos^2 \varphi(t)} \operatorname{sign}[\sin \varphi(t)] + r \sin \varphi(t) - \sqrt{l^2 - r^2} \operatorname{sign}[\sin \varphi(t)] \quad (3-21)$$

U jednadžbi (3-21) pojavljuje se izraz $sign[\sin\varphi(t)]$ jer rješenja korijena $\sqrt{1 - \left(\frac{r}{l}\right)^2 \cos^2\varphi(t)}$ i $\sqrt{l^2 - r^2}$ poprimaju pozitivne i negativne vrijednosti. Kada bi se uzimale samo pozitivne vrijednosti, put koji platforma prevali do krajnjeg gornjeg položaja bio bi veći od puta koji platforma prevali do krajnjeg donjeg položaja, što ne odgovara istini. Naime, ta dva puta moraju biti iste veličine, ali suprotnog predznaka. Budući da put $s(t)$ poprima negativne vrijednosti za vrijednosti kuta motora $\varphi(t)$ iz intervala $\varphi(t) \in [0, -\pi]$, znači da predznak ova dva korijena mora biti negativan. Na intervalu $\varphi(t) \in [0, -\pi]$ negativne vrijednosti ima sinus ovih kuteva, te je zato u jednadžbu ubačena funkcija $sign$.

Izraz (3-20) pojednostavljuje se koristeći relaciju:

$$\sqrt{1-x} = 1 - \frac{x}{2} - \frac{x^2}{8} - \dots \quad (3-22)$$

Te se dobije:

$$\sqrt{1 - \left(\frac{r}{l}\right)^2 \cos^2\varphi(t)} \approx 1 - \frac{1}{2} \left(\frac{r}{l}\right)^2 \cos^2\varphi(t) \quad (3-23)$$

Ako se na izraz (3-23) primijeni trigonometrijska relacija

$$\cos^2\varphi(t) = \frac{1}{2}[1 + \cos\varphi(t)] \quad (3-24)$$

dobiva se konačni izraz za put koji platforma prevali zaokretom motora za kut $\varphi(t)$:

$$s(t) = r\sin\varphi(t) + l \left\{ 1 - \frac{1}{4} \left(\frac{r}{l}\right)^2 [1 + \cos 2\varphi(t)] \right\} [sign(\sin\varphi(t))] - \sqrt{l^2 - r^2} sign(\sin\varphi(t)) \quad (3-25)$$

Put $s(t)$ može se izraziti u ovisnosti o zakretu platforme oko x osi i oko y osi:

$$\begin{aligned} S_x(t) &= a\sin\alpha(t), \\ S_y(t) &= a\sin\beta(t) \end{aligned} \quad (3-26)$$

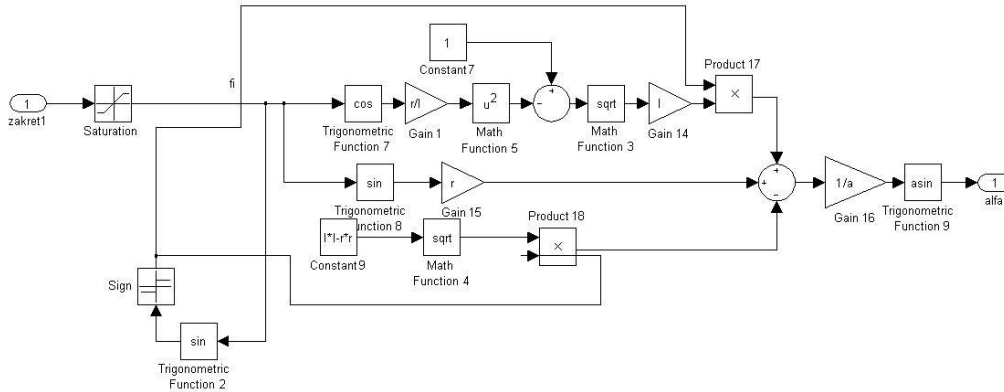
gdje je $a = 0.3 \text{ m}$ duljina, odnosno širina platforme.

Iz (3-25) i (3-26) slijede izrazi za proračun nagiba platforme oko x osi, odnosno oko y osi.

$$\alpha(t) = \arcsin\left(\frac{S_x(t)}{a}\right) \quad (3-27)$$

$$\beta(t) = \arcsin\left(\frac{S_y(t)}{a}\right) \quad (3-28)$$

Matematički podmodel platforme opisan jednačbama (3-25) do (3-28) odnosi se na savršeno ravnu krutu platformu. Model savršeno ravne krute platforme izrađen u Matlab Simulinku dan je slikom 3.6.



Slika 3.6: Model podsustava savršeno ravne krute platforme

3.1.3. Matematički model neravnina na platformi

Kao što je navedeno u tekstu korištena platforma u stvarnosti nije savršeno kruta i ravna. Iz tog razloga izrađen je matematički model neravnina na platformi. Što je matematički model sustava bolji postići će se bolja kvaliteta upravljanja na stvarnom sustavu.

Platforma je, kao što je navedeno, učvršćena u tri točke dok četvrti vrh nije učvršćen. Dva ruba platforme, koja se nalaze između tri učvršćena vrha platforme su ravna, dok su preostala dva savinuta u smjeru četvrtog, neučvršćenog vrha platforme. Savinuti rubovi platforme opisani su parabolnim jednačbama. Maksimalni progib platforme u slobodnom vrhu platforme, u odnosu na fiksni vrh platforme, iznosi 0.008 m, odnosno 8 mm.

Ploha platforme opisana je slijedećom jednačbom:

$$z = ax^2y + by^2x + c(x + y - \sqrt{x^2 + y^2}) \quad (3-29)$$

Kako bi se dobio nagib platforme u nekoj točki u smjeru x-osi, odnosno y-osi, potrebno je pronaći derivaciju jednačbe platforme u toj točki u željenom smjeru. Derivacija jednačbe (3-29) u smjeru x-osi, odnosno y-osi dana je slijedećim jednačbama:

$$\frac{\partial z}{\partial x} = 2axy + by^2 + c \left(1 - \frac{x}{\sqrt{x^2 + y^2}}\right) \quad (3-30)$$

$$\frac{\partial z}{\partial y} = 2bxy + ax^2 + c \left(1 - \frac{y}{\sqrt{x^2 + y^2}}\right) \quad (3-31)$$

Potrebno je postaviti sustav jednačbi kako bi se odredili koeficijenti a , b i c kojima bi se u potpunosti numerički odredila jednačba platforme. Vrijede slijedeće relacije:

$$\frac{\partial z}{\partial x_{x=y=0.3 m}} = \frac{\partial z}{\partial x_{x=y=0.3 m}} = -0.003 \text{ rad}$$

$$Z_{x=y=0.3 m} = -0.008 \text{ m}$$
(3-32)

Nakon uvrštavanja $x = 0.3$ i $y = 0.3$ u (3-29), (3-30) i (3-31), te uvažavanja uvjeta (2-32), dobije se sustav jednačbi:

$$\begin{aligned} 0.18a + 0.09b + 0.292893c + 0.003 &= 0 \\ 0.09a + 0.18b + 0.292893c + 0.003 &= 0 \\ 0.027a + 0.027b + 0.175736 + 0.008 &= 0 \end{aligned}$$
(3-33)

Rješavanjem navedenog sustava, dobiju se iznosi koeficijenata a, b i c :

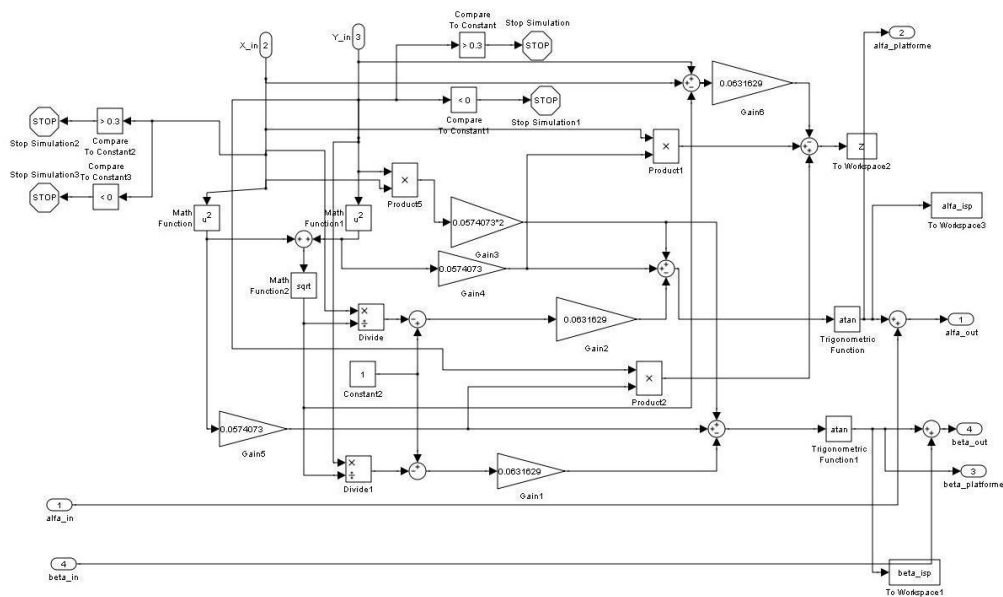
$$\begin{aligned} a &= b = 0.0574073 \\ c &= -0.0631629 \end{aligned}$$

Konačno, kut zakrivljenosti platforme u određenoj točki dan je sljedećim izrazom:

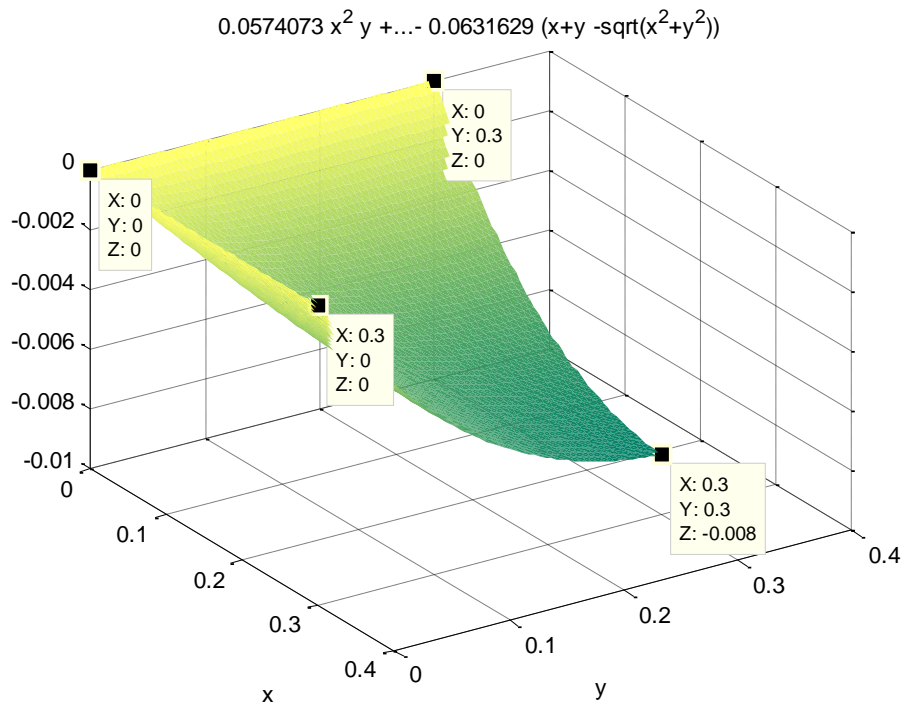
$$\alpha = \arctg\left(\frac{\partial z}{\partial x}\right)$$
(3-34)

$$\beta = \arctg\left(\frac{\partial z}{\partial y}\right)$$
(3-35)

Simulacijski model izrađen u Matlab Simulinku za podsustav neravnina na platformi dan je slikom 3.7 dok je slikom 3.8 dan graf plohe koja modelira zakrivljenosti platforme.



Slika 3.7: Model podsustava za proračunavanje zakrivljenosti platforme

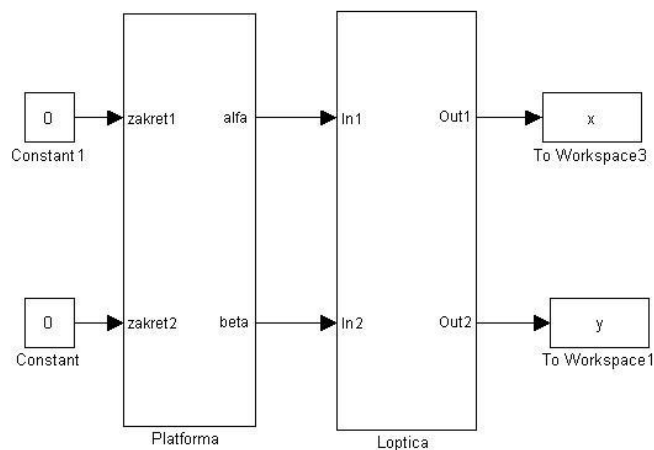


Slika 3.8: Ploha koja opisuje platformu

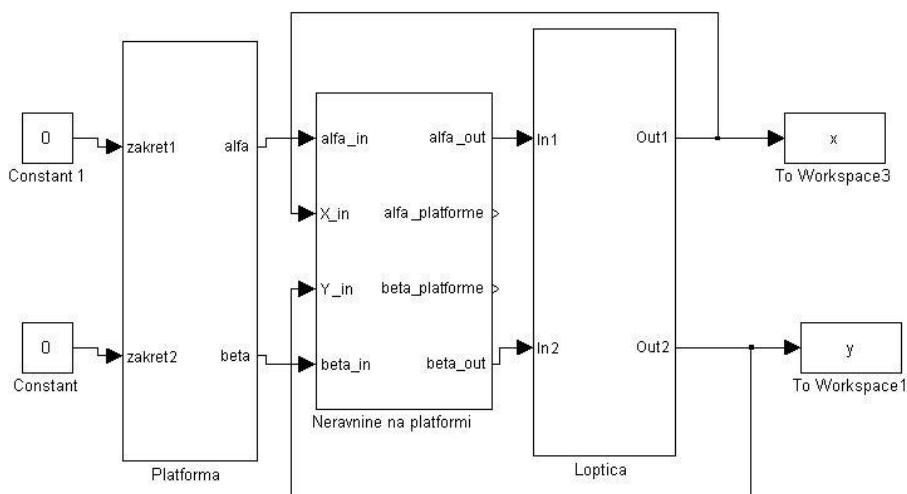
Model koji računa zakrivljenost platforme kao ulaz prima: kutove koji su izlaz podsustava s ravnom platformom te trenutnu poziciju loptice koju računa podsustav s matematičkim opisom loptice. Na osnovu trenutne pozicije loptice te uz matematički model neravnina na platformi računa se kut zakrivljenosti platforme. Vrijednost kuta zakrivljenosti platforme dodaje se vrijednosti kuta za ravnu platformu i tako se dobije stvarni kut trenutnog nagiba loptice. Podsustav kao izlaz daje dva kuta: jedan koji predstavlja pravi kut nagiba loptice te drugi koji predstavlja nagib zbog neravnina. Potonji se koristi za pomoćnu upravljačku petlju u svrhu linearizacije modela što će biti izneseno u poglavlju 3.2. U modelu koji računa zakrivljenost implementirano je zaustavljanje simulacije ukoliko loptica ispadne s platforme (tj. ukoliko vrijednost pozicije nije unutar intervala 0.0 do 0.3 m).

3.1.4. Provjera ispravnosti matematičkih modela

Određeni su matematički modeli koji u potpunosti opisuju ponašanje sustava, potrebno je provjeriti njihovu ispravnost. Testiranja su izvršena za model savršeno ravne i krute plohe te za model s neravninama. Ulaz u sustav je nula, odnosno, pretpostavlja se da je platforma u vodoravnom položaju i sve radi u otvorenoj petlji. Početna pozicija loptice za oba pokusa namještena je na ($x = 0.1 m, y = 0.05m$). Simulacijska shema izrađena u Simulinku za slučaj potpuno ravne plohe dana je na slici 3.9 dok je simulacijska shema za slučaj plohe s neravninama dana na slici 3.10.

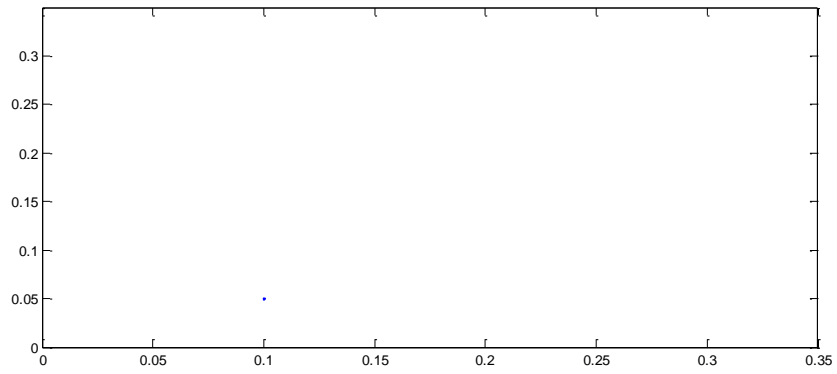


Slika 3.9: Simulacijska shema za savršeno ravnu i krutu platformu

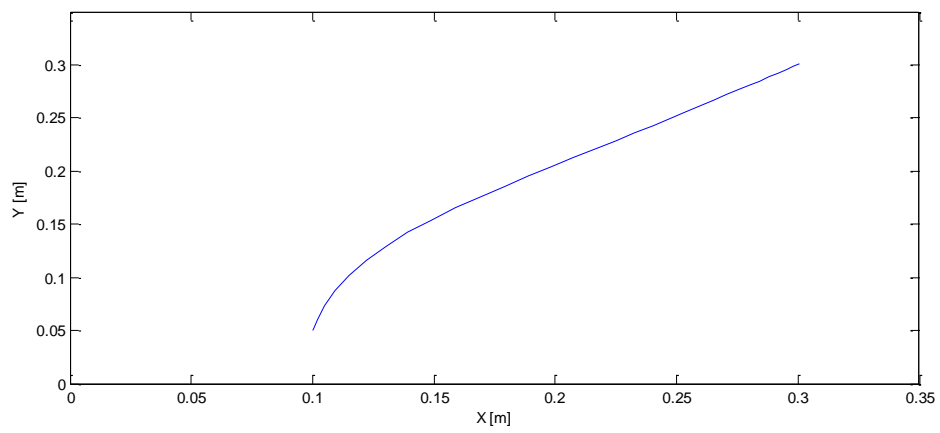


Slika 3.10: Simulacijska shema za platformu s neravninama

Odzivi za pomak po x i y osi dani su na slikama 3.11 i 3.12. Na slici 3.11 dan je odziv za potpuno ravno platformu. Primjećuje se samo jedna točka na poziciji (0.1,0.05) što znači da loptica stoji stabilno u toj točki. Taj rezultat je u potpunosti očekivan za ravnu platformu, bez pobude loptica stoji mirno na početnoj poziciji. Model za ravnu platformu u potpunosti je ispravan. Na slici 3.12 dan je odziv za platformu s neravninama. Loptica se počinje gibati iz početne pozicije prema rubu platforme zbog neravnina. Ovaj rezultat u potpunosti je očekivan te potvrđuje ispravnost modela za platformu s neravninama.



Slika 3.11: Pozicija loptice na savršeno ravnoj i krutoj platformi



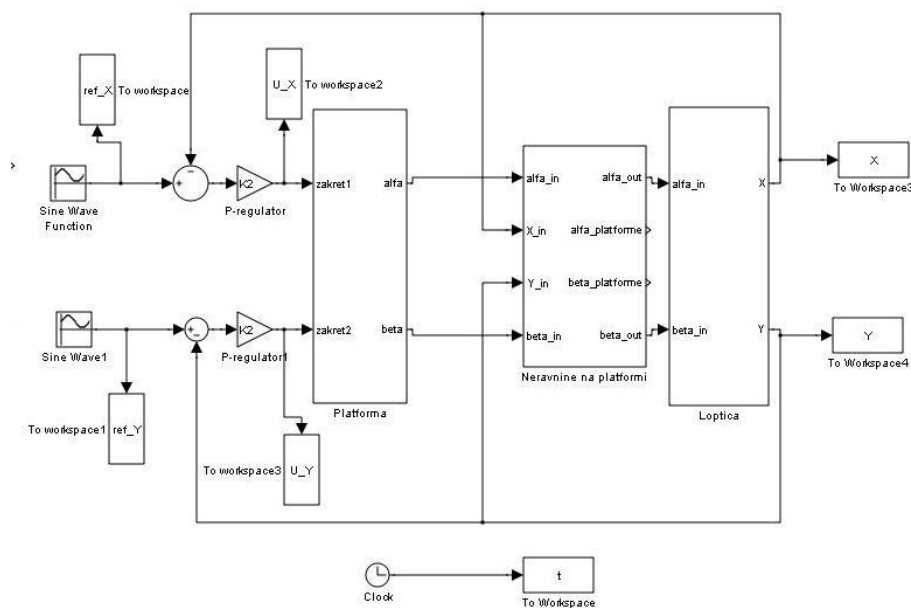
Slika 3.12: Pozicija loptice na platformi s neravninama

3.2. Sinteza algoritma upravljanja

Za promatrani sustav određeni su matematički modeli. Matematički modeli su provjereni te dovoljno vjerno oslikavaju stvarni fizički sustav koji se promatra. Korak koji slijedi je implementiranje metoda upravljanja na simulacijskim modelima. Metoda koja se prva koristila je zatvaranje upravljačkog kruga te dodavanje P regulatora. Rezultati nisu bili zadovoljavajući upravo zbog nelinearnosti platforme. Da bi se izbjegao utjecaj nelinearne platforme dodana je pomoćna upravljačka petlja koja ispravlja utjecaj neravnina na platformi (linearizacija u povratnoj vezi). S linearizacijom i uz P regulator dobili su se znatno bolji odzivi sustava ali ipak uz neke neželjene posljedice (oscilatoran odziv). Da bi se riješio taj problem analitičkom metodom sinteze regulatora „Truxal-Guillemín“ projektiran je regulator koji u potpunosti odgovara promatranom sustavu.

3.2.1. Upravljanje P regulatorom

Upravljanje P regulatorom je najjednostavnija metoda upravljanja u sustavima automatskog upravljanja. Karakterizira ga samo pojačanje u zatvorenom upravljačkom krugu. Koristi se za stabiliziranje procesa prvog reda. P regulator karakterizira pogrešku u stacionarnom stanju. Pojačanje regulatora određeno je eksperimentalno i to na modelu bez neravnina. Najbolje ponašanje dobiveno je za iznos pojačanja -6. Na slici 3.13 dan je model u Matlab Simulinku za upravljanje sustavom uz P regulator.

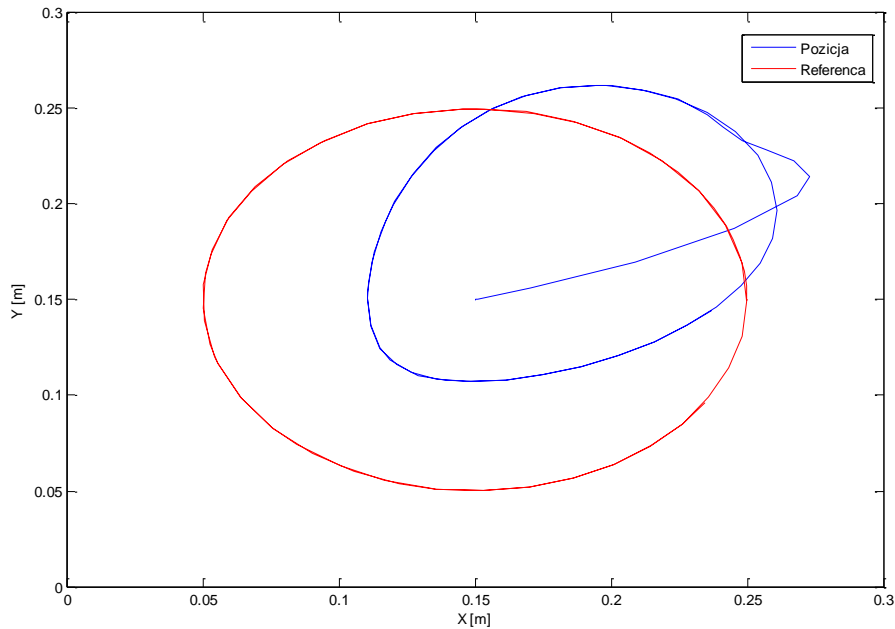


Slika 3.13: Simulacijska shema za platformu s neravninama i P regulatorom

3.2.1.1. Simulacija kružnice

Simulacija je provedena korištenjem Matlab Simulink sheme na slici 3.13. Kao referentni signal odabrana je sinusna pobuda oblika $r(t) = 0.1 \cdot \sin\left(\frac{1}{5}t + \frac{\pi}{2}\right) + 0.15$ za x os odnosno $r(t) = 0.1 \cdot \sin\left(\frac{1}{5}t\right) + 0.15$ za y os. Ta dva signala predstavljaju kružnicu s centrom u $(0.15\text{ m}, 0.15\text{ m})$ te radijusom 0.1 m . Početna pozicija loptice zadana je $(X = 0.15\text{ m}, Y = 0.15\text{ m})$. Na slici 3.14 dana je

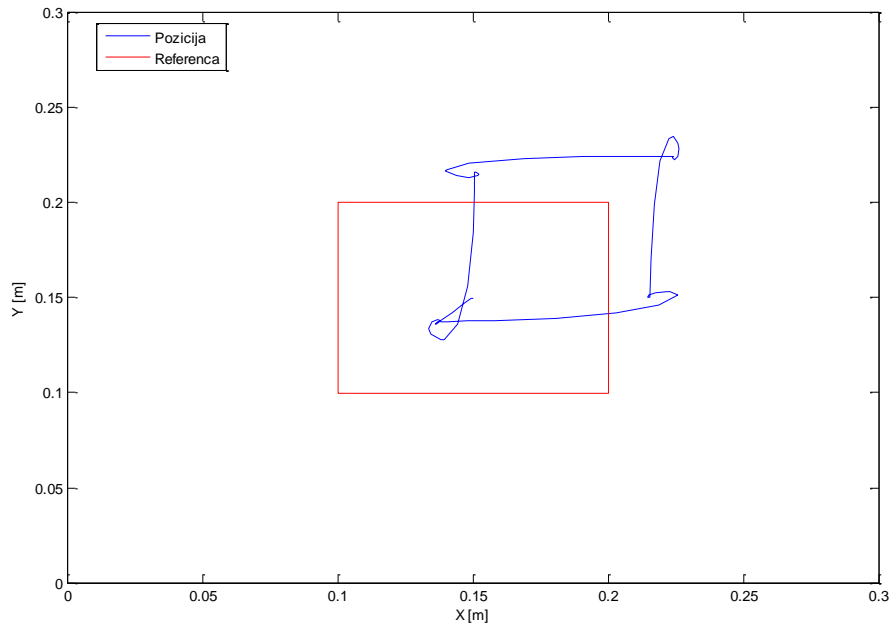
pozicija loptice na platformi. Vidi se da loptica ne izvodi pravu kružnicu koja u stvarnosti ima elipsoidni oblik te je pomaknuta prema četvrtom neučvršćenom vrhu platforme. Upravo je u tom smjeru platforma „savinuta“ te zbog toga ne dobivamo očekivani odziv. U prilogu A dani su odzivi x i y pozicije u odnosu na referentni signal te upravljački signali P regulatora za obje osi.



Slika 3.14: Pozicija loptice na platformi za simulaciju kružnice

3.2.1.2. Simulacija kvadrata

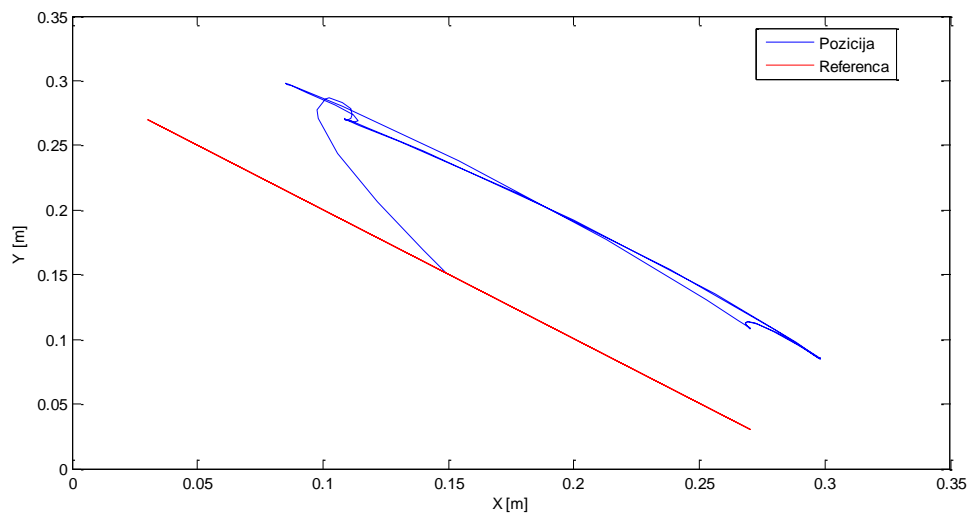
Simulacija je provedena korištenjem Matlab Simulink sheme na slici 3.13. uz drugačiji signal reference. Referenca je određena tako da signal opisuje kvadrat s vrhovima u $(0.1, 0.1)$ $(0.2, 0.1)$ $(0.2, 0.2)$ $(0.1, 0.2)$. Početna pozicija loptice zadana je $(X = 0.15 \text{ m}, Y = 0.15 \text{ m})$. Na slici 3.15 dana je pozicija loptice na platformi. Zbog platforme koja nije savršeno kruta i ravna „kvadrat“ se ne nalazi na svom mjestu već je pomaknut prema četvrtom neučvršćenom rubu platforme jer je u tom smjeru platforma zakrivljena. U prilogu A dani su odzivi x i y pozicije u odnosu na referentni signal te upravljački signali P regulatora za obje osi.



Slika 3.15: Pozicija loptice na platformi za simulaciju kvadrata

3.2.1.3. Simulacija pravca

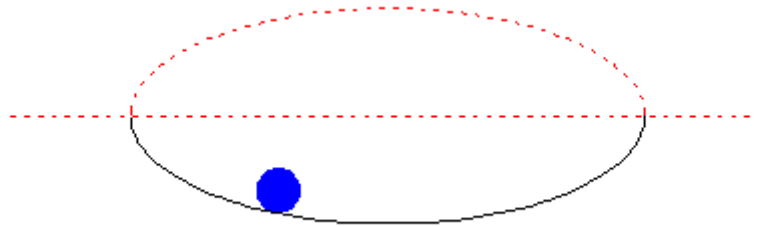
Simulacija je provedena korištenjem Matlab Simulink sheme na slici 3.13. uz drugačiji signal reference. Referenca je određena tako da signal opisuje dužinu s vrhovima u $(0.27, 0.03)$ $(0.03, 0.27)$. Početna pozicija loptice zadana je $(X = 0.15 \text{ m}, Y = 0.15 \text{ m})$. Na slici 3.16 dana je pozicija loptice na platformi. Zbog platforme koja nije savršeno kruta i ravna „dužina“ se ne nalazi na svom mjestu već je pomaknuta prema četvrtom neučvršćenom rubu platforme jer je u tom smjeru platforma zakrivljena. U prilogu A dani su odzivi x i y pozicije u odnosu na referentni signal te upravljački signali P regulatora za obje osi.



Slika 3.16: Pozicija loptice na platformi za simulaciju pravca

3.2.2. Pomoćna upravljačka petlja za linearizaciju platforme

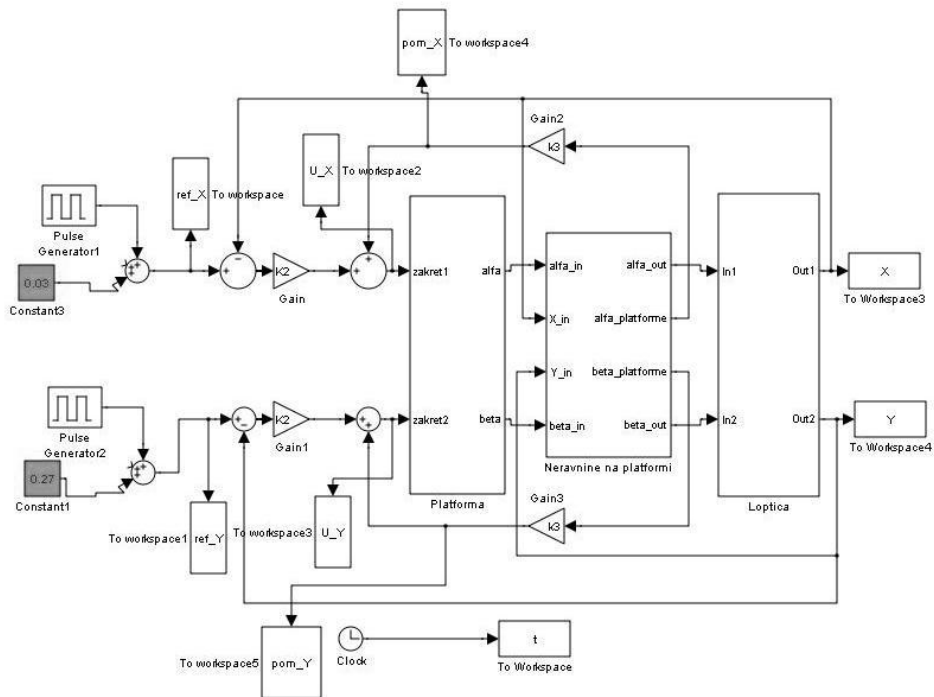
Kao što se vidi iz odziva u poglavlju 3.2.1 odzivi nisu zadovoljavajući za upravljanje P regulatorom. Razlog tome leži u činjenici da platforma u stvarnosti nije ravna. Da bi sustav imao dobre odzive i da bi njime mogli kvalitetno upravljati mora postojati mehanizam koji „ispravlja“ neravnine na platformi. Na slici 3.17 dan je shematski prikaz platforme s lopticom.



Slika 3.17: Shematski prikaz platforme s lopticom

Platforma je udubljena te zbog toga loptica dobiva ubrzanje čak i kad je platforma u potpuno horizontalnom položaju. Na slici 3.17 crnom bojom označena je platforma kakva je ona u stvarnosti. Da bi se izbjegli utjecaji neravnina uvodimo upravljanje po kutu zakrivljenosti. Pomoćna upravljačka petlja okreće platformu za kut zakrivljenosti platforme ali u suprotnom smjeru te uz eksperimentalno određeno pojačanje. Takvim upravljanjem lineariziramo platformu te ona ima oblik ravne isprekidane crte na slici 3.17. Kao što se vidi iz simulacijske sheme na slici 3.18 pomoćnom upravljačkom petljom izvedeno je upravljanje kutom zakreta platforme s obzirom na kut zakrivljenosti platforme uz pojačanje k_3 . Pojačanje k_3 određeno je eksperimentalno te ono iznosi -14.7931.

Ideja upravljanja po kutu zakrivljenosti platforme intuitivna je i jednostavna. Osnova ovakvog upravljanja čini dobro razrađen matematički model neravnina na platformi. Iako je rješenje upravljanja kuta motora a time kutom nagiba platforme po kutu zakrivljenosti neravnina jednostavno, kvaliteta upravljanja značajno je poboljšana što će biti pokazano odzivima u nastavku.



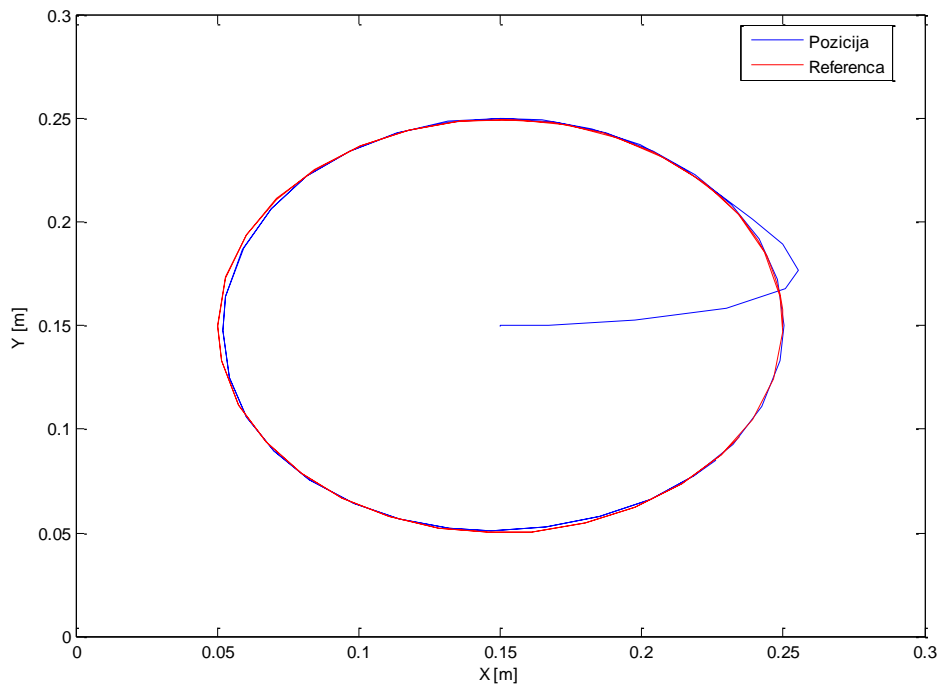
Slika 3.18: Simulacijska shema za lineariziranu platformu s neravninama i P regulatorom

3.2.2.1. Simulacija kružnice

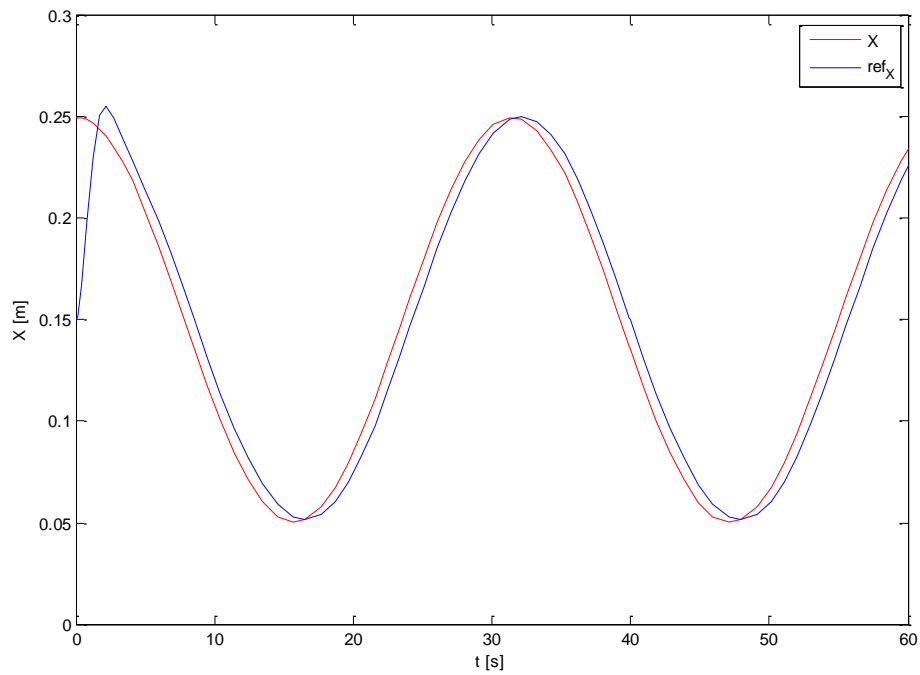
Simulacija je provedena korištenjem Matlab Simulink sheme na slici 3.18. Kao referentni signal odabrana je sinusna pobuda oblika $r(t) = 0.1 \cdot \sin\left(\frac{1}{5}t + \frac{\pi}{2}\right) + 0.15$ za x os odnosno $r(t) = 0.1 \cdot \sin\left(\frac{1}{5}t\right) + 0.15$ za y os. Ta dva signala predstavljaju kružnicu s centrom u $(0.15\text{ m}, 0.15\text{ m})$ te radijusom 0.1 m . Početna pozicija loptice zadana je $(X = 0.15\text{ m}, Y = 0.15\text{ m})$. Na slici 3.19 dana je pozicija loptice na platformi. Vidi se da pozicija loptice dobro prati zadanu referencu za razliku od simulacija provedenih bez linearizacije neravnina. Sustav se dobro ponaša upravo zbog korištenja linearizacije u pomoćnoj upravljačkoj petlji. Na slici 3.20 i 3.21 dan je odziv x i y pozicije u odnosu na referentni signal. Iz odziva se vidi da upravljana veličina (pozicija) dobro prati referentnu veličinu uz malo fazno kašnjenje jer sustav nije idealan. Na slikama 3.22 i 3.23 dan je upravljački signal na ulazu u sustav. Plavom linijom prikazan je ukupni upravljački signal dok je crvenom dan upravljački signal pomoćne upravljačke petlje za linearizaciju. Iz odziva se vidi da je utjecaj pomoćne upravljačke petlje na upravljački signal velik te da je upravo pomoćna petlja zaslužna za bolje odzive sustava. Koristeći istu Matlab Simulink simulacijsku shemu provedene su simulacije kvadrata i pravca. U prilogu B dani su kompletni odzivi navedenih simulacija; odzivi x i y pozicije u odnosu na referentni signal te upravljački signali za obje osi.

Simulacijski rezultati su slični onima za simulaciju kružnice, u slučaju kvadrata pozicija loptice dobro prati zadanu referencu. Kod simulacije pravca primijećene su nesavršenosti upravljanja jednostavnim P regulatorom. Upravljana veličina tj. pozicija loptice ima nadvišenje te ono uzrokuje stalno pomicanje loptice od željenog vrha dužine zbog velike dinamike sustava za dani referentni signal. Ovakvi rezultati

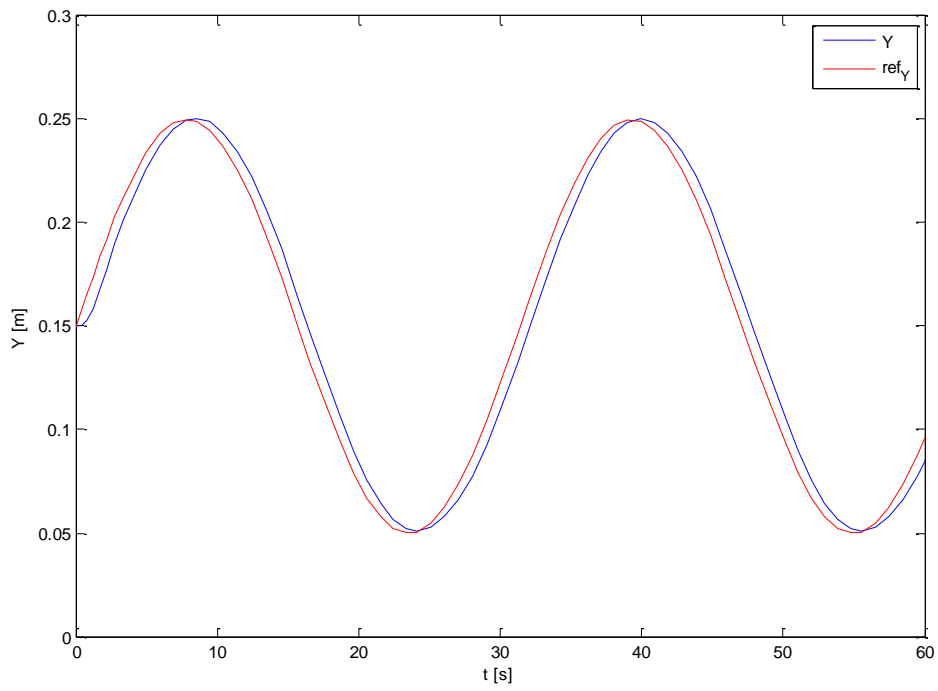
uvjetuje projektiranje regulatora koji će biti primjeren ovakvom sustavu te koji će imati bolju kvalitetu upravljanja.



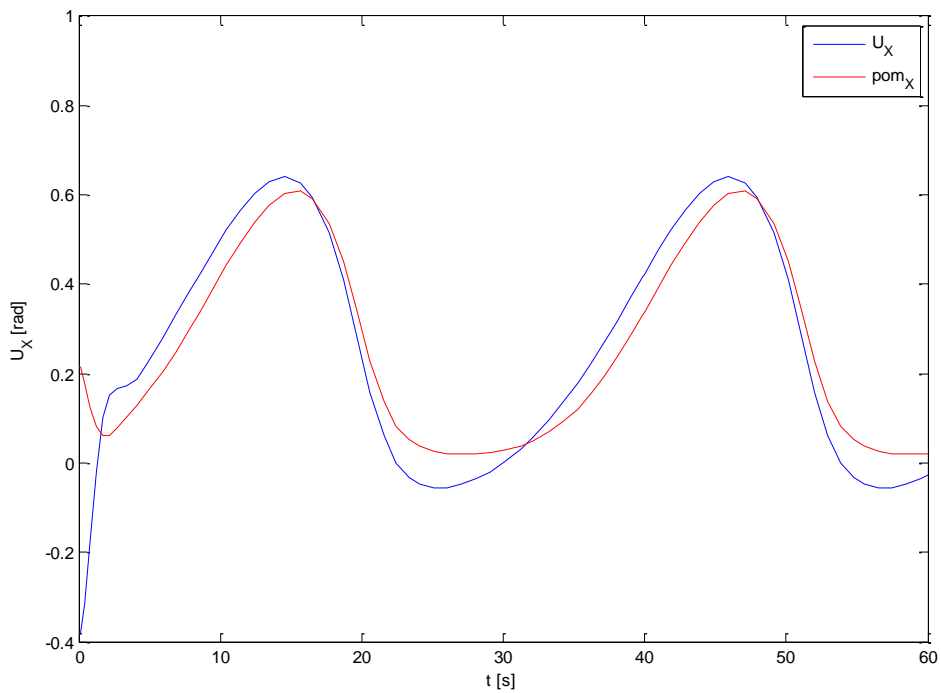
Slika 3.19: Pozicija loptice na platformi za simulaciju kružnice



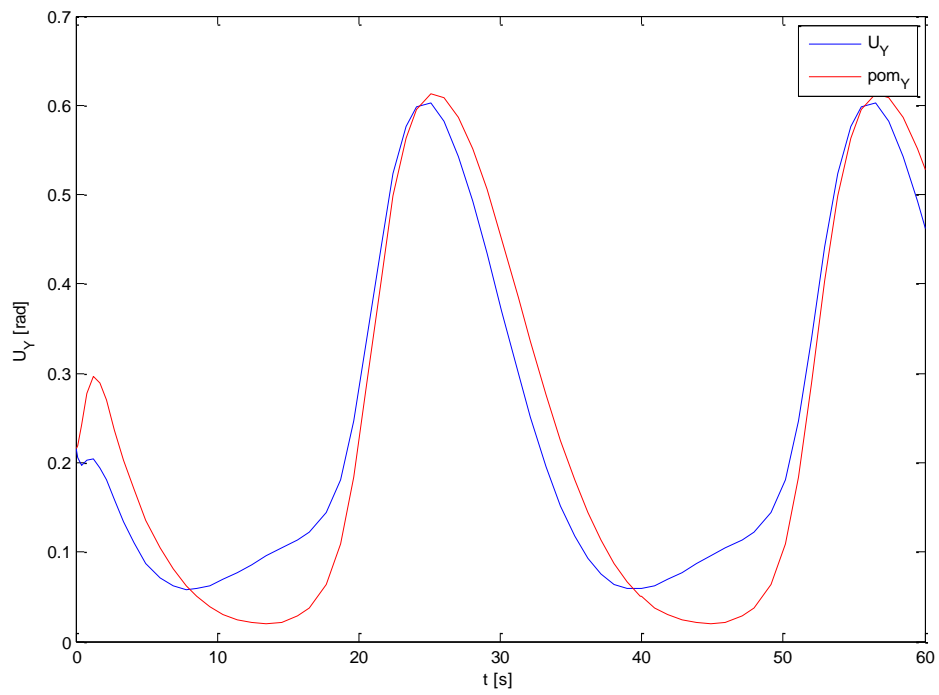
Slika 3.20: Pozicija loptice po x osi u odnosu na referentni signal



Slika 3.21: Pozicija loptice po y osi u odnosu na referentni signal



Slika 3.22: Upravljački signal na ulazu u sustav za x os



Slika 3.23: Upravljački signal na ulazu u sustav za y os

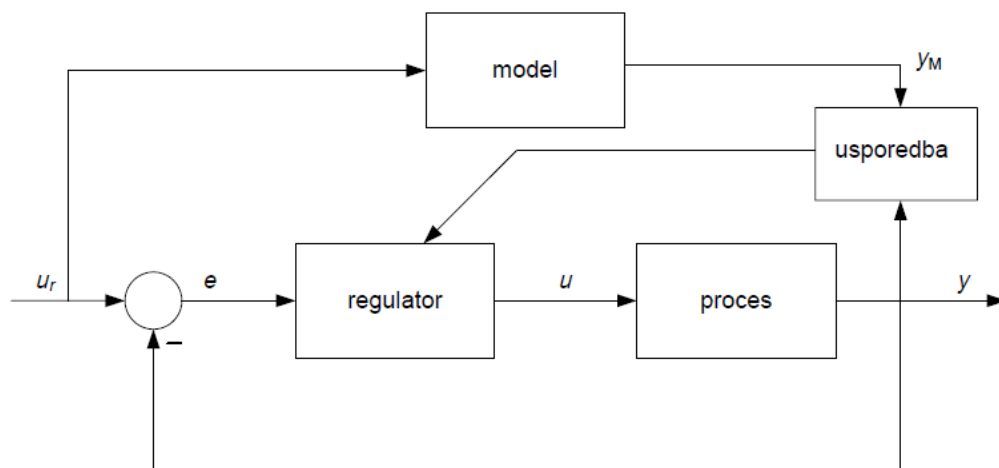
3.3. Projektiranje regulatora metodom Truxal-Guillemin

Poglavlje 3.2 služilo je kao uvod u projektiranje pokaznog regulatora koji će udovoljiti svim traženim pokazateljima kvalitete. Ti pokazatelji u potpunosti će biti udovoljeni regulatorom projektiranim Truxal-Guillemin-ovim postupkom. Određeni su matematički modeli sustava, riješen je problem nelinearnosti platforme te se na temelju tih rješenja pristupa sintezi željenog regulatora.

Prema [14] Truxal-Guillemin je analitički postupak sinteze jer je polazište pri sintezi zatvoreni regulacijski krug opisan željenom prijenosnom funkcijom:

$$G_r(s) \stackrel{!}{=} G_m(s) \quad (3-36)$$

Gdje je $G_m(s)$ željena (modelska) prijenosna funkcija zatvorenog sustava (referentni model). Modelska funkcija proizlazi iz specifikacija kakvoće koja se obično zasniva na odgovarajućoj prijelaznoj funkciji $h_r(t)$. Uz poznato vladanje procesa i uz odabrani $G_m(s)$ može se tada neposredno projektirati odgovarajući regulator. Na slici 3.24 dan je regulacijski krug koji se razmatra pri sintezi Truxal-Guilleminovim postupkom.



Slika 3.24: Promatrani regulacijski krug pri sintezi Truxal-Guillemin-om [15]

Proces $G_s(s)$ opisan je racionalnom prijenosnom funkcijom:

$$G_s(s) = \frac{D(s)}{C(s)} = \frac{d_0 + d_1s + \dots + d_ms^m}{c_0 + c_1s + \dots + c_ns^n} \quad (3-37)$$

uz pretpostavke da polinomi brojnika $D(s)$ i nazivnika $C(s)$ nemaju zajedničke korijene; proces je stabilan i posjeduje svojstva minimalne faze; stupanj nazivnika $n >$ stupnja nazivnika m .

Prijenosna funkcija regulatora kojega treba projektirati:

$$G_R(s) = \frac{B(s)}{A(s)} = \frac{b_0 + b_1s + \dots + b_ws^w}{a_0 + a_1s + \dots + a_zs^z} \quad (3-38)$$

pri tome mora biti ispunjen uvjet realizacije ($z \geq w$).

$G_m(s)$ može, uz uvjet realizacije regulatora, imati proizvoljan oblik:

$$G_m(s) = \frac{\alpha(s)}{\beta(s)} = \frac{\alpha_0 + \alpha_1 s + \dots + \alpha_v s^v}{\beta_0 + \beta_1 s + \dots + \beta_u s^u} \quad (3-39)$$

Regulator treba tako projektirati da prijenosna funkcija zatvorenog sustava odgovara modelskoj prijenosnoj funkciji $G_m(s)$. Iz blok sheme na slici 3.24 dobije se prijenosna funkcija s obzirom na referentnu vrijednost:

$$G_r(s) = \frac{G_R(s)G_s(s)}{1 + G_R(s)G_s(s)} \stackrel{!}{=} G_m(s) \quad (3-40)$$

Iz čega slijedi da je prijenosna funkcija regulatora:

$$G_R(s) = \frac{1}{G_s(s)} \frac{G_m(s)}{1 - G_m(s)} \quad (3-41)$$

Mora biti ispunjeno da je stupanj brojnika regulatora $B(s) = w = n + v \leq$ stupanj nazivnika $A(s) = z = u + m$, odakle slijedi

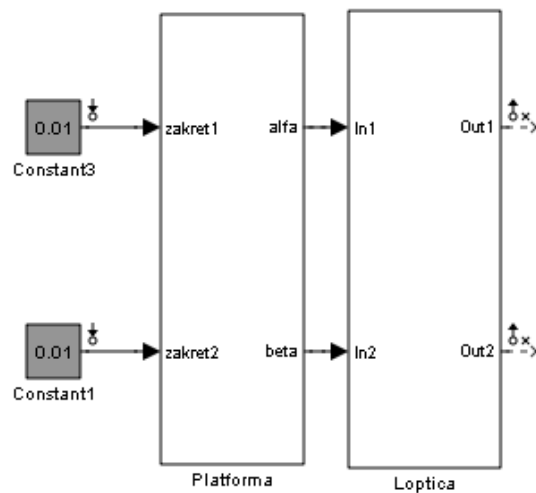
$$u - v \geq n - m. \quad (3-42)$$

Polni višak ($u-v$) željene prijenosne funkcije $G_m(s)$ mora biti veći (ili jednak) polnom višku ($n-m$) prijenosne funkcije procesa $G_s(s)$.

U prijenosnoj funkciji regulatora (3-41) pojavljuje se $G_s(s)^{-1}$. Dakle regulatorom bi se moglo potpuno kompenzirati dinamičko vladanje procesa. Stoga se projektirani regulator naziva kompenzacijskim regulatorom.

3.3.1. Linearizacija matematičkog modela sustava

Za potrebe sinteze regulatora potrebno je dobiti prijenosnu funkciju sustava tj. potrebno ga je linearizirati. Proces linearizacije dug je i računski zahtjevan te je zbog jednostavnosti linearizacija izvršenu u Simulinku korištenjem alata „Control and Estimation Tools manager“. Shema korištena u Simulinku dana je slikom 3.25.

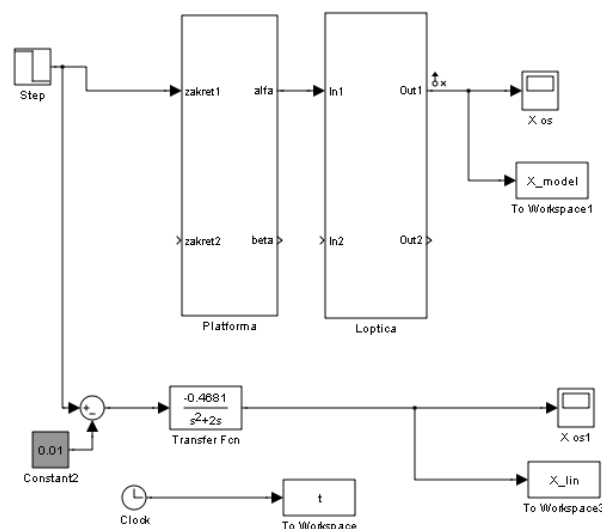


Slika 3.25: Simulink model korišten za linearizaciju

Zakret platforme odabran je kao ulazni signal, pozicija optice kao izlaz i to u otvorenoj petlji. Ulazna konstanta 0.01 m označava točku oko koje se vrši linearizacija. Nakon što je provedena linearizacija dobivena je prijenosna funkcija sustava tj. procesa:

$$G_s(s) = \frac{-0.4681}{s(s + 2)} \quad (3-43)$$

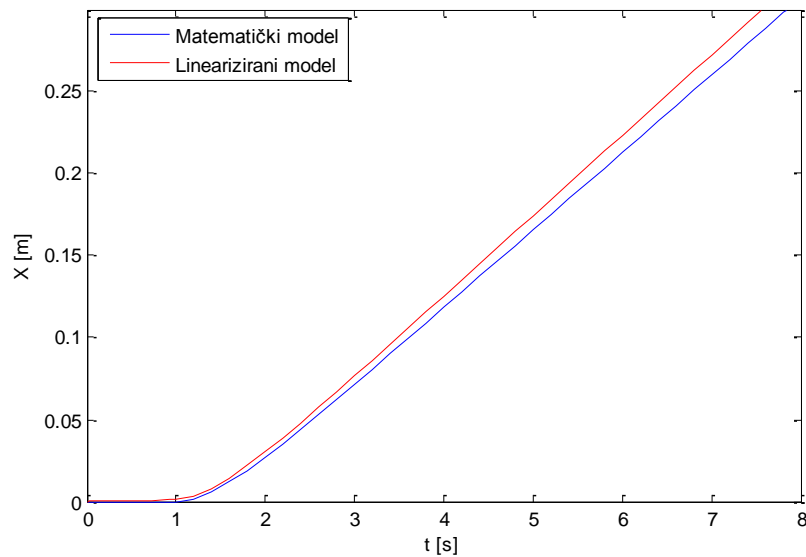
Da bi se provjerila ispravnost lineriziranog modela korištena je shema u Simulinku prikazana slikom 3.26.



Slika 3.26: Simulink model korišten za provjeru ispravnosti lineariziranog modela

Dobiven je odziv prikazan slikom 3.27. prema odzivu se vidi da je linearizirani model dovoljno dobar. Točnost aproksimacije lineariziranog modela povećava se što je model bliže zadanoj radnoj točki (0.01

m) što se vidi iz slike 3.27. Prijenosna funkcija dobivena linearizacijom u Simulinku može se koristiti kod sinteze regulatora.



Slika 3.27: Odzivi dobiveni simulacijskom shemom za usporedbu matematičkog i lineariziranog modela

3.3.2. Sinteza regulatora metodom Truxal-Guillemin

U poglavlju 3.3.1 dobivena je prijenosna funkcija procesa te ona glasi:

$$G_s(s) = \frac{-0.4681}{s(s+2)} \quad (3-44)$$

Proces je drugog reda te da bi se zadovoljio uvjet realizacije regulatora (3-42) odabire se modelska funkcija drugog reda čiji opći oblik glasi:

$$G_m(s) = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \quad (3-45)$$

Parametri modelske prijenosne funkcije slijede iz pokazatelja kvalitete, u konkretnom slučaju o nadvišenju te vremenu prvog maksimuma. Veza je dana sljedećim relacijama:

$$\sigma_m[\%] = 100 \cdot \exp\left(-\frac{\zeta\pi}{\sqrt{1-\zeta^2}}\right) \quad (3-46)$$

$$t_m = \frac{\pi}{\omega_n\sqrt{1-\zeta^2}} \quad (3-47)$$

Nakon niza eksperimenata na modelu odabrano je $\sigma_m[\%] = 1$, te $t_m = 2s$ jer je dobiveni regulator za te vrijednosti pokazao najbolje vladanje. Iz relacija (3-46) te (3-47) dobije se:

$$\zeta = 0.8261; \omega_n = 2.7873.$$

Uvrštavajući vrijednosti u (3-45) modelska funkcija dobiva oblik:

$$G_m(s) = \frac{7.769}{s^2 + 4.605s + 7.769} \quad (3-48)$$

Prijenosna funkcija regulatora na temelju (3-41) te uzimajući u obzir (3-44) te (3-48) iznosi:

$$G_R(s) = \frac{1}{G_s(s)} \frac{G_m(s)}{1 - G_m(s)} = \frac{-16.6s - 33.2}{s + 4.605} \quad (3-49)$$

Pri projektiranju regulatora kao ispomoć korišten je sljedeći programski kod u Matlabu:

```
%ulaz = nadvisenje(nad) i vrijeme prvog maximuma(tm)
zeta=-log(nad/100)/(sqrt(pi*pi+(log(nad/100))^2))
omega=pi/(tm*sqrt(1-zeta*zeta))
num=[omega*omega]
den=[1 2*omega*zeta omega*omega]
Gm=tf(num,den)
Gr2=Gm/(1-Gm)
Gr=Gr2*(1/f)
Gr=minreal(Gr)
Gr
```

Regulator dobiven Truxal-Guillemin postupkom potrebno je prebaciti iz kontinuirane u diskretnu domenu radi implementacije na digitalnom računalu. Korišten je ZOH postupak uz vrijeme diskretizacije $T_d = 0.1$. Vrijeme diskretizacije odabrano je sukladno dinamici sustava i izvedivosti na digitalnom računalu. Iz dobivene prijenosne funkcije u diskretnoj domeni implementira se jednadžbu diferencija. Korišten je idući odsječak koda u Matlabu:

```
>> Gr

Transfer function:
-16.6 s - 33.2
-----
s + 4.605

>> Gr_z=c2d(Gr,0.1,'zoh')

Transfer function:
-16.6 z + 13.94
-----
z - 0.631
```

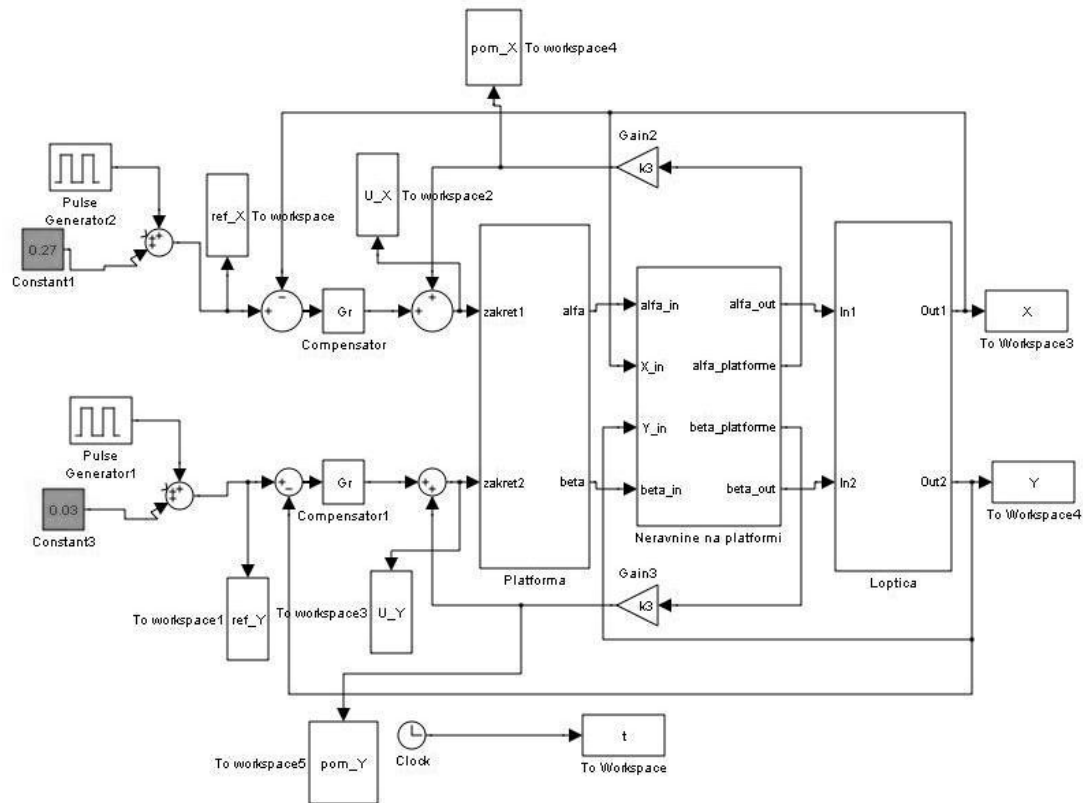
Iz dobivene prijenosne funkcije lako se dobije jednadžba diferencija:

$$u(k) = -16.6e(k) + 13.94e(k-1) + 0.631u(k-1) \quad (3-50)$$

3.3.3. Upravljanje projektiranim regulatorom

Kao što je objašnjeno u poglavlju 3.3.2 projektiran je kompenzacijski regulator koristeći Truxal-Guillemin metodu sinteze. Konačni model sustava s navedenim upravljanjem dan je slikom 3.28. Koristi se upravljanje projektiranim kompenzacijskim regulatorom na koje se dodaje upravljački signal

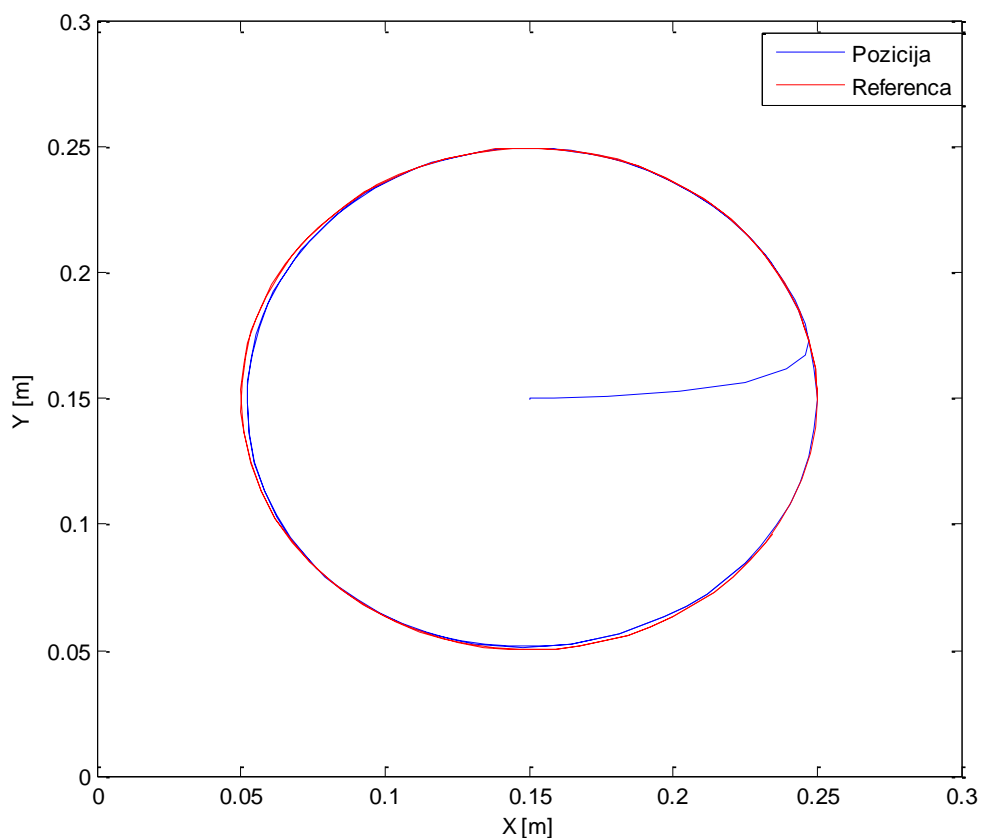
pomoćne upravljačke petlje. Upravljački signal zbroj je signala s kompenzacijskog regulatora i signala pomoćne upravljačke petlje koji se tvori po kutu zakrivljenosti platforme. Model na slici 3.28 korišten ja za kasnije navedene simulacije. Iz odziva danih u nastavku vidi se da projektirani regulator u kombinaciji s pomoćnom upravljačkom petljom udovoljava svim kriterijima kvalitete upravljanja. Dobiveni odziv nije oscilatoran, nema nadvišenja te gotovo nema pogreške u stacionarnom stanju.



Slika 3.28: Simulacijska shema za upravljanje sustavom koristeći kompenzacijski regulator i pomoćnu upravljačku petlju po kutu zakrivljenosti platforme

3.3.3.1. Simulacija kružnice

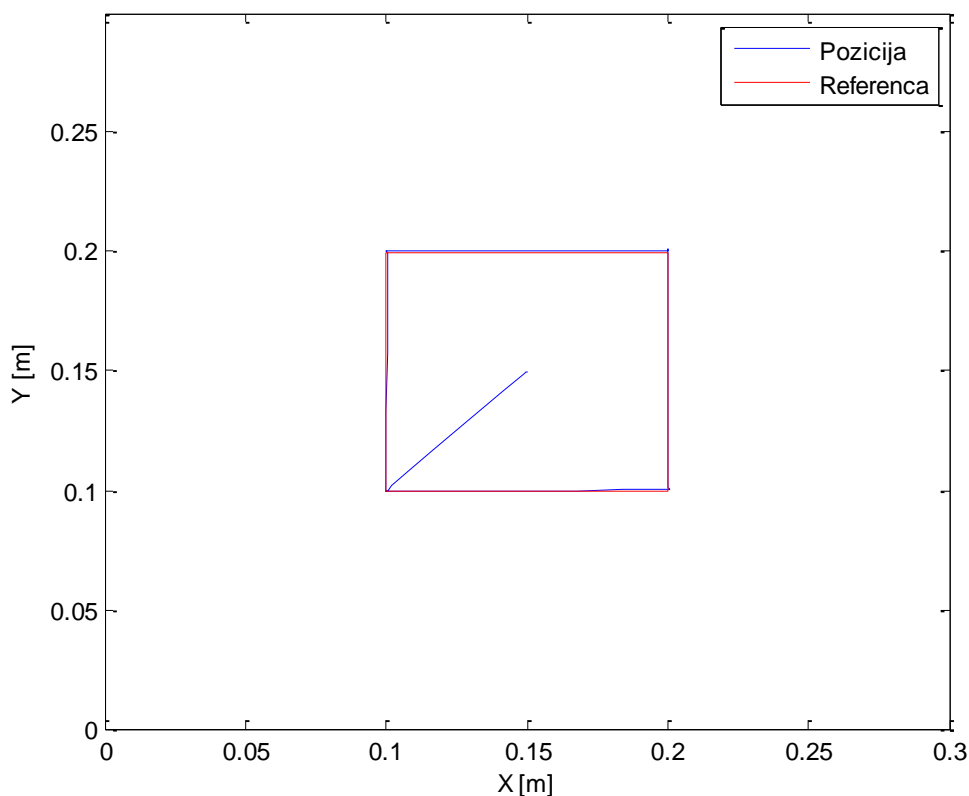
Simulacija je provedena korištenjem sheme na slici 3.28. Kao referentni signal odabrana je sinusna pobuda oblika $r(t) = 0.1 \cdot \sin\left(\frac{1}{5}t + \frac{\pi}{2}\right) + 0.15$ za x os odnosno $r(t) = 0.1 \cdot \sin\left(\frac{1}{5}t\right) + 0.15$ za y os. Ta dva signala predstavljaju kružnicu s centrom u $(0.15\text{ m}, 0.15\text{ m})$ te radijusom 0.1 m . Početna pozicija loptice zadana je $(X = 0.15\text{ m}, Y = 0.15\text{ m})$. Na slici 3.29 dana je pozicija loptice na platformi. Vidi se da pozicija loptice dobro prati zadanu referencu. U prilogu C dani su odzivi x i y pozicije u odnosu na referentni signal te upravljački signali projektiranog regulatora za obje osi.



Slika 3.29: Pozicija optice na platformi za simulaciju kružnice

3.3.3.2. Simulacija kvadrata

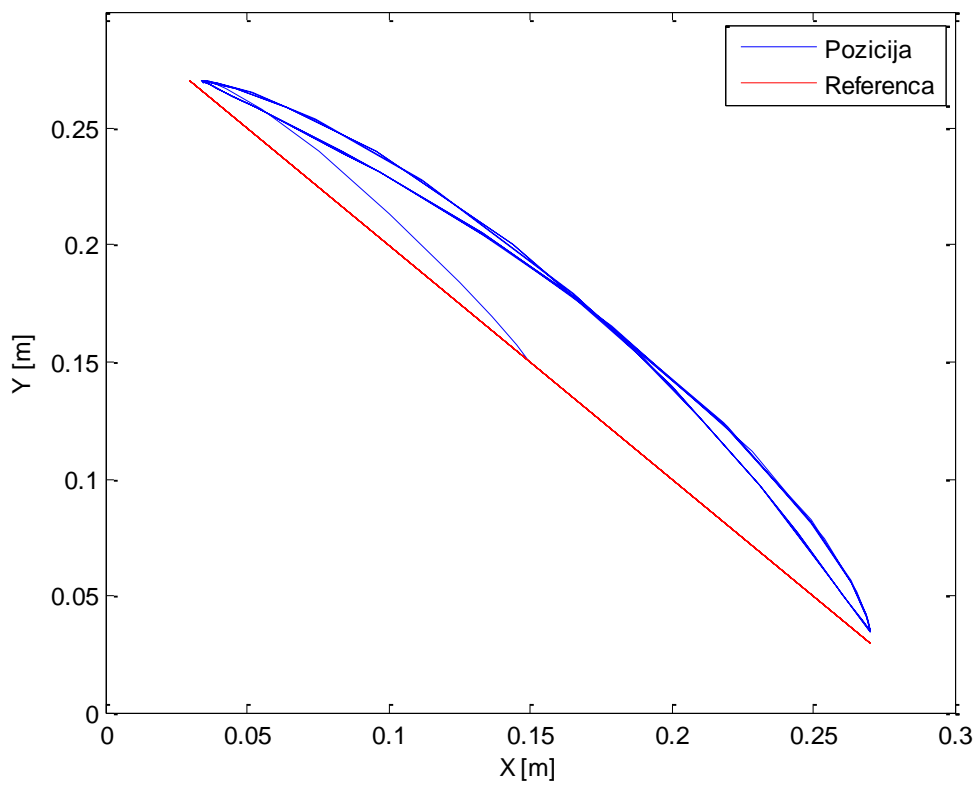
Simulacija je provedena korištenjem sheme na slici 3.28. uz drugačiji signal reference. Referenca je određena tako da signal opisuje kvadrat s vrhovima u $(0.1, 0.1)$ $(0.2, 0.1)$ $(0.2, 0.2)$ $(0.1, 0.2)$. Početna pozicija optice zadana je $(X = 0.15 \text{ m}, Y = 0.15 \text{ m})$. Na slici 3.30 dana je pozicija optice na platformi. Iz odziva se vidi da se pozicija optice gotovo savršeno podudara s referentnim signalom. U prilogu C dani su odzivi x i y pozicije u odnosu na referentni signal te upravljački signali projektiranog regulatora za obje osi.



Slika 3.30: Pozicija loptice na platformi za simulaciju kvadrata

3.3.3.3. Simulacija pravca

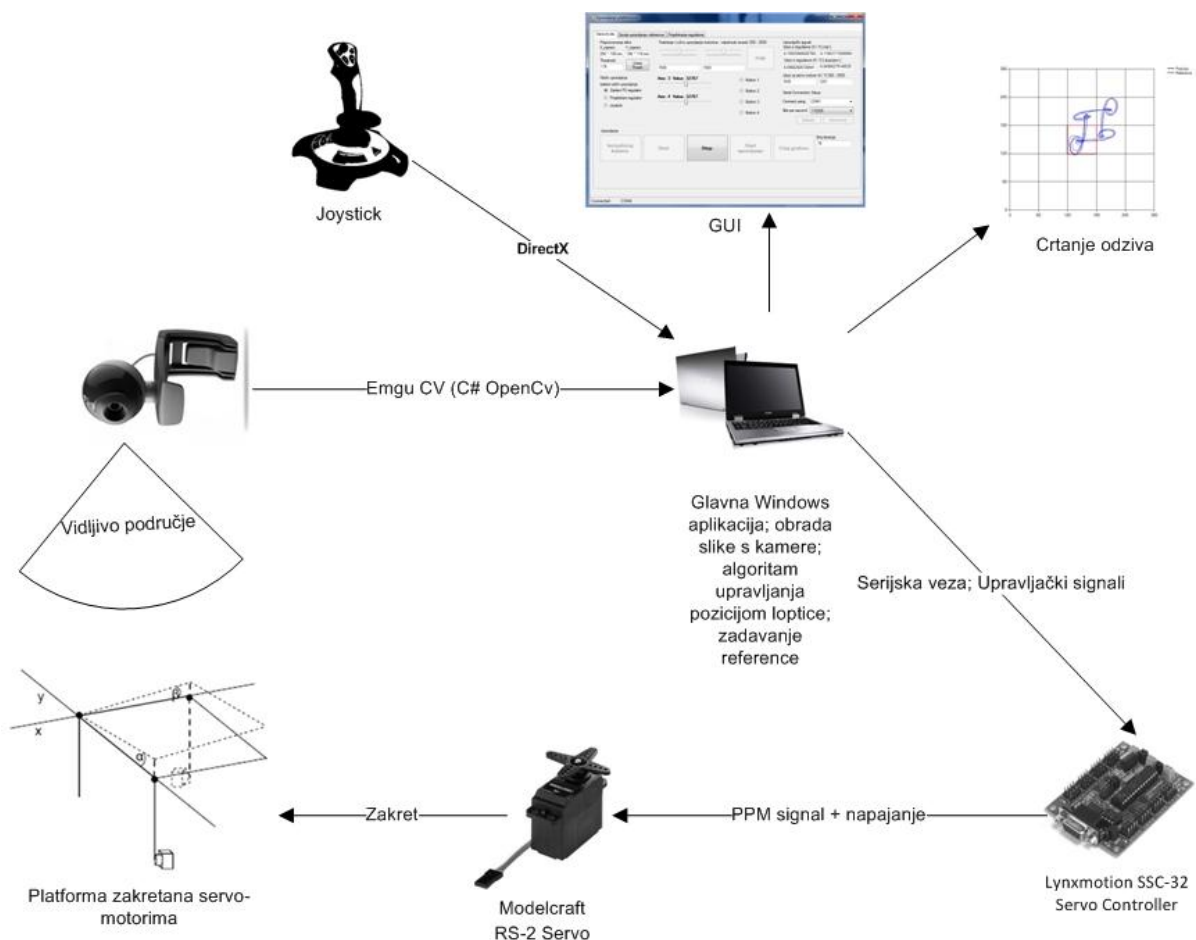
Simulacija je provedena korištenjem sheme na slici 3.28. uz drugačiji signal reference. Referenca je određena tako da signal opisuje dužinu s vrhovima u $(0.27, 0.03)$ $(0.03, 0.27)$. Početna pozicija loptice zadana je $(X = 0.15 \text{ m}, Y = 0.15 \text{ m})$. Na slici 3.31 dana je pozicija loptice na platformi. Na odzivu se vidi da upravljana veličina (pozicija) ne prati savršeno referentnu vrijednost, zbog nelinearnosti postoji regulacijsko odstupanje koje kao rezultat daje pravac koji je pomaknut prema neučvršćenom rubu platforme. Ipak usporedi li se ovaj odziv s onim dobivenim upravljanjem s jednostavnim P regulatorom vidi se da ipak projektirani regulator u kombinaciji s pomoćnom upravljačkom petljom daje puno bolju kvalitetu upravljanja. U prilogu C dani su odzivi x i y pozicije u odnosu na referentni signal te upravljački signali projektiranog regulatora za obje osi.



Slika 3.31: Pozicija loptice na platformi za simulaciju pravca

3.4. Realizirani sustav

Esencijalni dio ovog rada predstavlja izradu infrastrukture koja se očituje u programskoj i sklopovskoj podršci za implementaciju i provjeru simuliranih algoritama upravljanja na stvarnom sustavu. Bitna odlika realiziranog sustava je da je u potpunosti samostalan, može se koristiti na bilo kojem osobnom računalu s Windows operativnim sistemom. Na osobno računalo postavljaju se minimalni zahtjevi, ono mora imati RS 232 serijsko sučelje te 2 USB priključka. Osobno računalo ne mora imati veliku procesorsku moć pošto se kod izrade sustava veliku pozornost pridavalo jednostavnosti i brzini korištenih algoritama. Ovakav sustav predstavlja odličan edukacijski model za učenje i testiranje kako jednostavnih klasičnih algoritama upravljanja tako i naprednih algoritama upravljanja. Načelna shema realiziranog sustava dana je slikom 3.32.



Slika 3.32: Shematski prikaz realiziranog sustava

Sustav se sastoji od: platforme pokretane dvama servomotorima, web-kamerom postavljenom okomito iznad platforme, kontrolerom za servomotore, osobnog računala na kojem se izvodi Windows aplikacija te igraća palica (*eng.* Joystick) koja služi za ručno upravljanje.

Glavna aplikacija izvodi se na osobnom računalu s Windows operativnim sustavom. Povratna veza sustava ostvaruje se preko web-kamere koja je na računalo spojena preko USB sučelja. Glavna aplikacija hvata sliku s kamere te iz nje izvlači informaciju o trenutnoj poziciji loptice na platformi. Istovremeno, u glavnoj aplikaciji izvodi se upravljački algoritam koji na temelju razlike željene (tj.

referentne) i stvarne pozicije kuglice računa odgovarajuće zakrete servomotora kako bi kuglica dostigla željenu tj. referentnu vrijednost. Informacija o zakretu šalje se preko serijske veze kontroleru. Kontroler, na čije izlaze su spojeni servomotori, na osnovu podataka primljenih preko serijske veze šalje upravljačke signale servomotorima. Servomotori se zakreću se za odgovarajuću vrijednost te uzrokuju promjenu kuta nagibe platforme. Promjenom kuta nagiba platforme mijenjaju se sile koje djeluju na lopticu te se ona uslijed tih sila giba po platformi. Promjenu pozicije kuglice registrira kamera te se opisani proces ponavlja sve dok je omogućen u glavnoj aplikaciji.

3.4.1. Platforma i loptica

Fizički dio sustava sastoji se, kao što je navedeno, od nelinearne platforme pokretane servomotorima, kontrolera za servomotore te web-kamere.

Platforma, prikazana slikom 3.33, izrađena je od crnog, plastičnog, sjajnog materijala. Ima oblik kvadrata dimenzija 0.3×0.3 [m]. Jedan vrh platforme je učvršćen zglobnim mehanizmom koji platformi omogućava podizanje i spuštanje oko točke vrha platforme. Učvršćeni vrh predstavlja ishodište koordinatnog sustava pozicije loptice na platformi. Vrh na suprotnom kraju slobodan je te se nesmetano spušta i diže oko metalne vodilice koja služi kao stabilizator gibanja platforme. Preostala dva kraja platforme pričvršćena su na vratilo servomotora preko bregastog mehanizma koji pretvaraju rotacijsko gibanje vratila servomotora (pogonski član) u translacijsko gibanje platforme (vođeni član).

Ploha platforme uslijed korištenja i gibanja loptice poprimila je oštećenja koja se očituju u sitnim rupicama i udubinama koje stvaraju problem pri upravljanju. Uslijed nesavršene mehaničke konstrukcije platforme, te plastičnog materijala od kojeg je izrađena, platforma nije savršeno ravna i kruta ploha već je savinuta po sredini u smjeru slobodnog vrha platforme. Savinutost platforme uzeta je obzir pri analizi sustava te je i modelirana plohom parabolnog oblika što je objašnjeno u poglavlju 3.1.3.

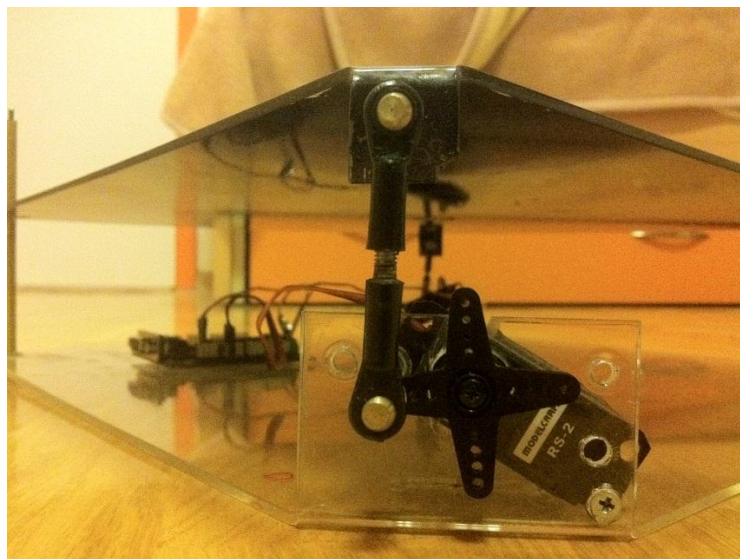


Slika 3.33: Platforma s lopticom

Na površini platforme giba se loptica. Korištena je metalna kuglica mase $m_l = 0.03 \text{ kg}$ te polumjera $R = 0.0115 \text{ m}$. Izrazito bitna karakteristika korištene kuglice je njena glatka površina. Što je kuglica glatkija i „savršenija“ biti će postignuto bolje upravljanje jer je utjecaj možebitnih neravnina na samoj kuglici smanjen na minimum.

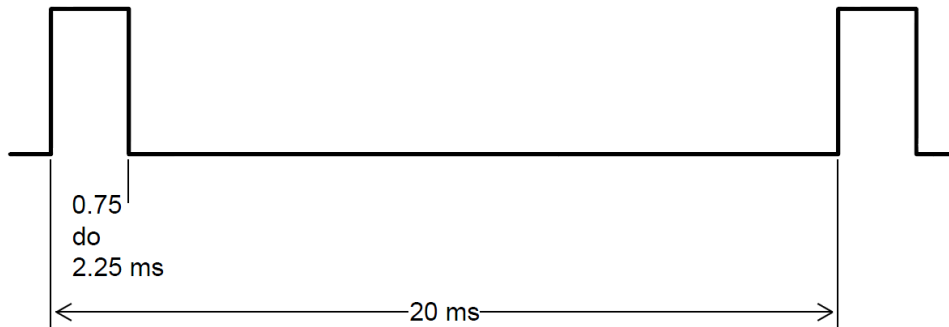
3.4.2. Sklopovlje

Platformu pokreću dva Modelcraft RS-2 motora (slika 3.34). Ovakav tip aktuatora klasičan je servomotor koji se većinom koristi za školske modele robota, RC hobi modele aviona, brodova i slično, shodno tome cijena ovakvih motora nije visoka. Servomotori rade pri naponu napajanja od 4.6 V do 6 V . Upravlja se u otvorenoj petlji, dakle ne postoji informacija o dostizanju zadane pozicije. Unaprjeđenje sustava svakako bi predstavljala ugradnja servomotora s povratnom informacijom tj. koji bi se mogao upravljati u zatvorenoj petlji.



Slika 3.34: Korišteni Modelcraft RS-2 motor s pripadajućim bregastim mehanizmom

Pozicijom osovine servomotora upravlja se dovođenjem PPM (*eng. Pulse Position Modulation*) signala na njegove ulaze. Oblik PPM signala dan je slikom 3.35. Period signala iznosi 20 ms , a sastoji se od pozitivnog dijela impulsa iznosa napona napajanja U_{cc} , u trajanju od 0.75 ms do 2.25 ms , dok je ostatak periode prisutan negativni dio signala. Variranjem dužine trajanja impulsa upravlja se pozicijom osovine servomotora. Dužina impulsa od 0.75 ms označava krajnji lijevi položaj osovine, dužina od 1.5 ms označava srednji položaj osovine servomotora dok dužina impulsa od 2.25 ms označava krajnji desni položaj osovine servomotora. Na osnovu primljenog signala servomotor postavlja osovinu u zadani položaj. Upravljačka petlja pozicije osovine servomotora zatvorena je elektroničkim sklopovljem unutar servomotora, te, kao što je već navedeno, korisnik servomotora nema pristup toj povratnoj informaciji.



Slika 3.35: Oblik PPM signala

Upravljački PPM signali servomotora generiraju se korištenjem Lynxmotion SSC-32 Servo Controllera. Navedeni kontroler, prikazan slikom 3.36, koristi se pri upravljanju hobi i školskim modelima robota. Servomotorima osigurava izvor napajanja te generiranje PPM signala.



Slika 3.36: Lynxmotion SSC-32 Servo Controller [16]

Kontroler ima mogućnost spajanja do 32 servomotora. Jezgru kontrolera čini Atmel ATMEGA168-20PU mikrokontroler. Kao ulaz ima serijsko sučelje. Preko serijskog sučelja prima naredbe, koje moraju biti zadane u predviđenom formatu, te na svoje izlaze daje odgovarajuće PPM signale koji upravljaju spojenim servomotorima. Na službenoj Internet stranici proizvođača [16] mogu se pronaći detaljne upute o spajanju i korištenju kontrolora no najvažniji je način upravljanja spojenim servomotorima. Kontroler preko serijskog sučelja očekuje podatak u idućem formatu:

```
"#26 P1200 #8 P2300 \r");
```

Podatci #26 i #8 označavaju upravljanje servomotorima spojenima na 26. i 8. Kanal kontrolera. Podatci P1200 odnosno P2300 označavaju željenu poziciju osovine servomotora. Pozicijom se može upravljati u rasponu od P500 (krajnji lijevi položaj osovine servomotora) do P2500 (krajnji desni položaj osovine servomotora).

Vizualna povratna veza realizirana je korištenjem web-kamera: Logitech Webcam C200. Izgled kamere dan je slikom 3.37. Web-kamera se na osobno računalo spaja preko USB sučelja. Kamera ima mogućnost hvatanja slike u stvarnom vremenu u VGA rezoluciji tj. veličina slike dobivene u pikselima s kamere je 640×480 . Ovakav tip kamere odabran je zbog dostupnosti, jednostavnosti korištenja te niske cijene. Kamera se fiksira na visinu od otprilike 0.6 m od platforme. Kamera je okomita na platformu koja joj se nalazi u vidljivom području.



Slika 3.37: Logitech Webcam C200 [17]

3.5. Programska podrška

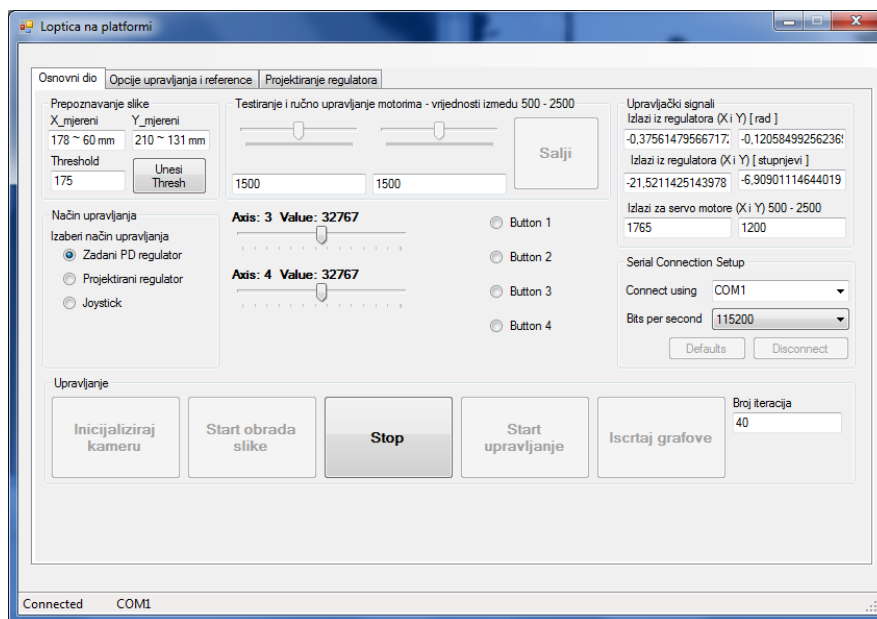
Programska podrška sastoji se od glavne aplikacije u kojoj su realizirane sve bitne funkcionalnosti sustava. Glavna aplikacija temelj je realiziranog sustava pošto ista omogućava upravljanje te projektiranje i ispitivanje novih tipova algoritama bez potrebe za bilo kojim drugim programskim alatom (npr. Microsoft Visual Studio ili neki drugi alat koji služi razvoju i izvođenju različitih programskih odsječaka). Realizirana aplikacija omogućava samostalno i nezavisno korištenje sustava loptice na platformi kao edukacijskog ili istraživačkog modela.

Razvijena aplikacija napisana je u programskom jeziku C# uz upotrebu .NET 3.5 programskih biblioteka, a kao razvojno okruženje korišten je Microsoft Visual Studio 2008. Za izradu grafičkog sučelja aplikacije korištene su Windows Forms biblioteke koje su dio .NET Frameworka. U glavnoj aplikaciji realizirane su iduće funkcionalnosti:

- Dohvat i obrada slike s web-kamere (prepoznavanje točne pozicije loptice na platformi).
- Upravljanje servomotorima preko serijske veze.
- Dodavanje novih algoritama upravljanja sustavom.
- Zadavanje referentnog položaja loptice u stvarnom vremenu.
- Ručno upravljanje sustavom putem igrace palice ili klizača u aplikaciji.
- Crtanje odziva pozicije loptice na platformi i upravljačkih signala algoritma upravljanja.
- Promjena parametara osnovnog realiziranog PD regulatora u stvarnom vremenu.

3.5.1. Grafičko korisničko sučelje

Korisničko sučelje sastoji se od glavne aplikacije pokretane u Windows operativnom sustavu. Korisničko sučelje omogućava korisniku pokretanje ili zaustavljanje upravljanja sustavom, mijenjanje različitih postavki i parametara sustava, nadziranje rada sustava, unošenje i izvođenje novih algoritama upravljanja te pregledavanje i spremanje odziva sustava. Izgled realizirane glavne aplikacije dan je slikom 3.38. Glavna aplikacija podijeljena je na 3 kartice. Prva osnovna kartica sadrži gumbe za upravljanje radom sustava, izbornik načina upravljanja, osnovne informacije koje korisniku služe za nadzor nad sustavom (pozicija loptice, izlazi iz pojedinih upravljačkih algoritama, klizače i pokazivače koji pokazuju informacije dobivene s igrace palice) te postavke serijske veze. Grafičko korisničko sučelje je izrađeno vodeći se načelom jednostavnosti, preglednosti i intuitivnosti.



Slika 3.38: Izgled razvijene Windows aplikacije

3.5.1.1. Upravljački dio

Upravljački dio razvijene aplikacije omogućava upravljanje sustavom loptice na platformi. Upravljački dio dobro je objašnjen blok dijagramom na slici 3.39. Ukoliko je upravljanje sustavom loptice na platformi programski uključeno aktivira se *Timer* kontrola. Unutar opcija *Timer* kontrole postavljen je period od 0.1 s pošto je to period diskretizacije koji odgovara dinamici upravljanog sustava. Ukoliko je *Timer* kontrola aktivna ona generira *dogadjaj Timer_tick* (eng. event) koji se izvodi svako unaprijed postavljenog perioda. Upravljački dio dakle izvodi se na svaki *tick Timera*.

Nakon pokretanja funkcije upravljački dio iz algoritma obrade slike preuzima se informacija o trenutnoj poziciji loptice na platformi te se računaju trenutne vrijednosti reference i regulacijskog odstupanja:

$$e(k) = r(k) - y(k) \quad (3-51)$$

Izračunom osnovnih vrijednosti upravljanja sustava, ovisno o izboru načina upravljanja korisnika, računaju se upravljački signali iz osnovnog PD regulatora ili korisnički projektiranog regulatora ili se uzimaju s igraće palice. S dobivenim upravljačkim podacima postoje svi podatci koji kompletno opisuju strukturu upravljanja te se shodno tome podatci o poziciji i upravljački signali upisuju u *log* datoteke. *Log* datoteke koriste se kao izvor podataka za crtanje grafova odziva.

Upravljački signali (bez obzira na način upravljanja) izraženi su u radijanima. Kontroler ne prima podatke o zakretu spojenih servomotora u radijanima već u predodređenom formatu (poglavlje 3.4.2). Podatci iz radijana moraju se normirati u podatke koje prima kontroler. Empirijski su određene funkcije normiranja upravljačkih signala u vrijednosti koje prima kontroler:

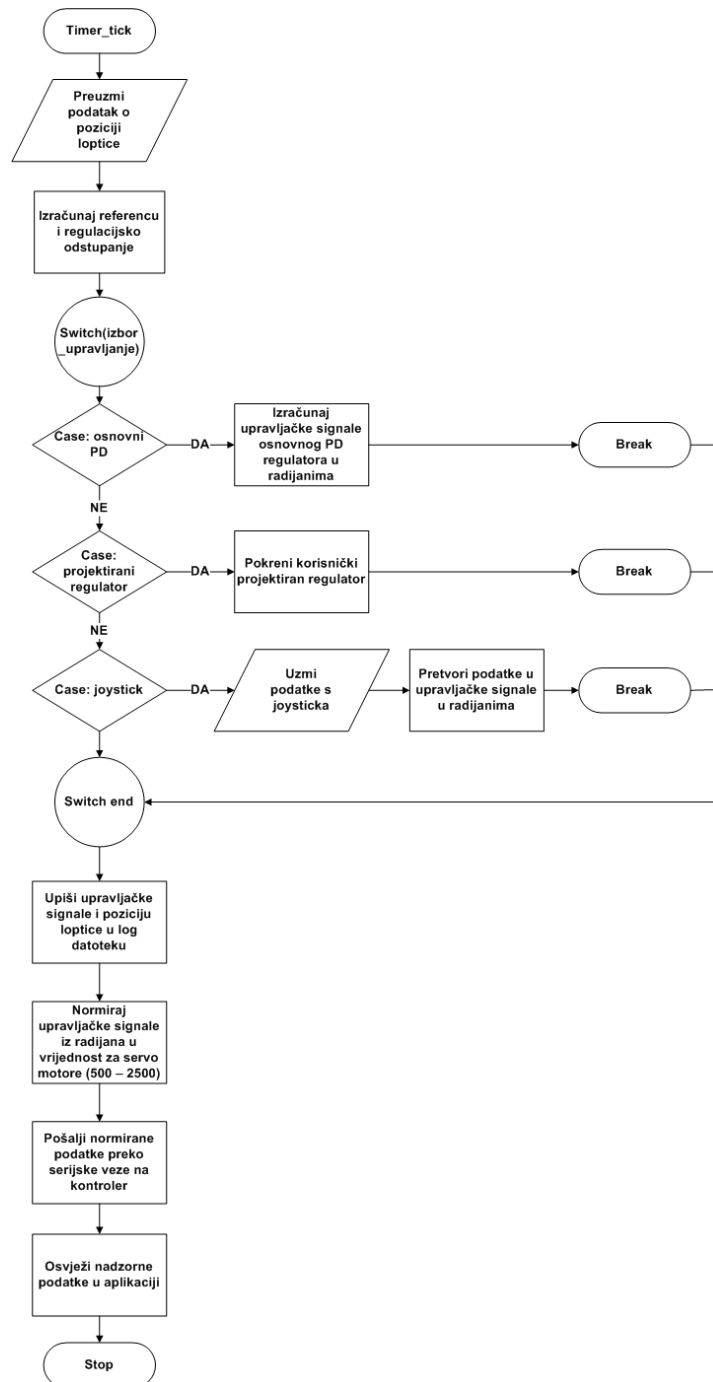
$$\begin{aligned} u_{servo_x} &= 1550 - u_x(k) * \left(\frac{1700}{\pi}\right); & u_x(k) &\in \left[0, \frac{\pi}{2}\right] \\ u_{servo_x} &= 1550 - u_x(k) * \left(\frac{1800}{\pi}\right); & u_x(k) &\in \left[-\frac{\pi}{2}, 0\right) \end{aligned} \quad (3-51)$$

Odnosno za upravljačke signale y osi:

$$u_{servoy} = 1270 + u_Y(k) * \left(\frac{1900}{\pi}\right); \quad u_X(k) \in \left[0, \frac{\pi}{2}\right]$$

$$u_{servoy} = 1270 + u_X(k) * \left(\frac{1800}{\pi}\right); \quad u_X(k) \in \left[-\frac{\pi}{2}, 0\right)$$
(3-52)

Normirani upravljački signali pretvoreni su u vrijednost koju zahtjeva kontroler. Normirani podatci se grupiraju te se u odgovarajućem predodređenom formatu preko serijske veze šalju na kontroler. Kontroler primi podatak preko serijske veze te pokreće servomotore.



Slika 3.39: Blok dijagram upravljačkog dijela

3.5.1.2. Serijska veza

Serijska veza esencijalni je dio sustava pošto je ona veza između programske podrške i sklopovlja koje pokreće cijeli sustav. Parametri i format poruka serijske veze predodređeni su tehničkim specifikacijama Lynxmotion SSC-32 Servo Controllera danima u [16]. Prema uputama i tehničkim specifikacijama kontrolera brzina prijenosa podataka preko serijske veze namješta se manipulacijom kratkospojnika na kontroleru. Izabrana je brzina od 115200 *bit/s*.

Serijska veza na osobnom računalu u glavnoj aplikaciji realizirana je korištenjem *SerialPort* kontrole. Inicijalizacija serijske veze izvodi se s parametrima priloženim u tablici 3.1.

Tablica 3.1: Parametri inicijalizacije serijske veze

Parametar	Vrijednost
<i>PortName</i>	Određuje korisnik – obično COM1
<i>BaudRate</i>	Određuje korisnik – preporučeno 115200 <i>bit/s</i>
<i>Parity</i>	Bez pariteta
<i>StopBits</i>	1
<i>Handshake</i>	Bez <i>handshakea</i>

3.5.1.3. Upravljanje igraćom palicom

Ručno upravljanje sustavom omogućeno je korištenjem igraće palice (*eng. joystick*) koja je na osobno računalo spojena preko USB sučelje. Komunikacija između glavne aplikacije i igraće palice ostvarena je korištenjem *Microsoft DirectX software development kit (SDK)* biblioteka te primjera Microsoft Visual Studio projekta napisanog u programskom jeziku C# dostupnog na [18]. Projekt [18] daje primjer kako ostvariti komunikaciju igraće palice i aplikacije na osobnom računalu.

Implementiran je razred `Joystick` koji predstavlja igraću palicu spojenu na računalo. Razred pruža podršku za igraću palicu koja ima do 6 osi upravljanja. Unutar razreda sadržane su sve varijable koje daju informacije o trenutnom stanju svih osi i gumba igraće palice: `axisA`, `axisB`, `axisC`, `axisD`, `axisE`, `Buttons` te varijabla `axisCount` koja sadržava broj osi prisutnih na spojenoj igraćoj palici. Razred sadrži metode za hvatanje, pronalaženje spojenih igraćih palica, otpuštanje igraće palice operativnom sustavu, konstruktor razreda te najvažnija metoda `UpdateStatus()`. `UpdateStatus()` metoda koristi se za osvježavanje podataka o trenutnom stanju osi i pritisnutih gumba na igraćoj palici. Pozivanjem navedene metode u varijable stanja razreda `Joystick` spremaju se nove informacije s igraće palice.

Pri pokretanju glavne aplikacije, ukoliko je na računalo spojena igraća palica, stvori se odgovarajući razred koji ju predstavlja. Stanja dviju glavnih osi igraće palice skupa s četiri glavna gumba mogu se vidjeti u stvarnom vremenu preko implementiranih klizača u glavnoj aplikaciji.

Vrijednosti osi igraće palice nalaze se u rasponu od 0 do 65535, 0 predstavlja krajnji lijevi položaj dok 65535 predstavlja krajnji desni. Upravljanje servomotorima izvodi se u radijanima, sukladno tome

vrijednosti dobivene s igraće palice potrebno je normirati. Normiranje se izvodi idućim linearnim relacijama:

$$\begin{aligned} u_x &= \left(\frac{\pi}{65535}\right) * joystick.Axis1 - \frac{\pi}{2} \\ u_y &= \left(\frac{\pi}{65535}\right) * joystick.Axis2 - \frac{\pi}{2} \end{aligned} \quad (3-53)$$

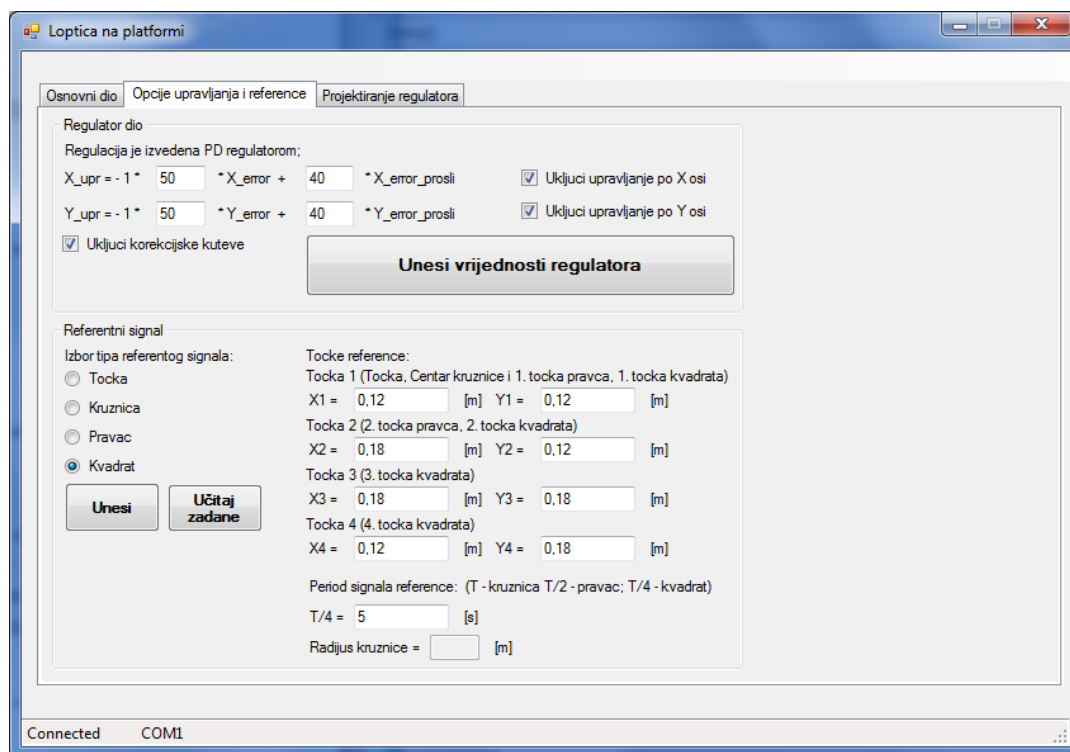
Vrijednosti dobivene navedenim normiranjem prosljeđuju se kontroleru kako je opisano u poglavlju 3.5.1.1.

3.5.1.4. Zadavanje referentnog položaja loptice

Zadavanje referentnog položaja loptice na platformi u glavnoj aplikaciji izvodi se pozivom funkcije `zadaj_referencu()`. Funkcija sadrži varijablu t koja predstavlja realno vrijeme izvođenja eksperimenta.

Korisnik preko grafičkog korisničkog sučelja (slika 3.40) može izabrati 4 različita tipa referentnog signala: točka, kružnica, pravac te kvadrat. Za sve različite tipove referentnih signala korisnik može unositi parametre signala. Tako korisnik može mijenjati vrijednost točke koju hvata loptica, centar i radijus kružnice, točke pravca i kvadrata te pripadajuće periode signala.

Funkcija `zadaj_referencu()` poziva se pri svakom pokretanju upravljačkog dijela tj. pri svakom koraku uzorkovanja sustava. Funkcija, uzevši u obzir izbor korisnika i unesene vrijednosti parametara signala, kao izlaz daje vrijednost referentne pozicije loptice za trenutni diskretizacijski korak.



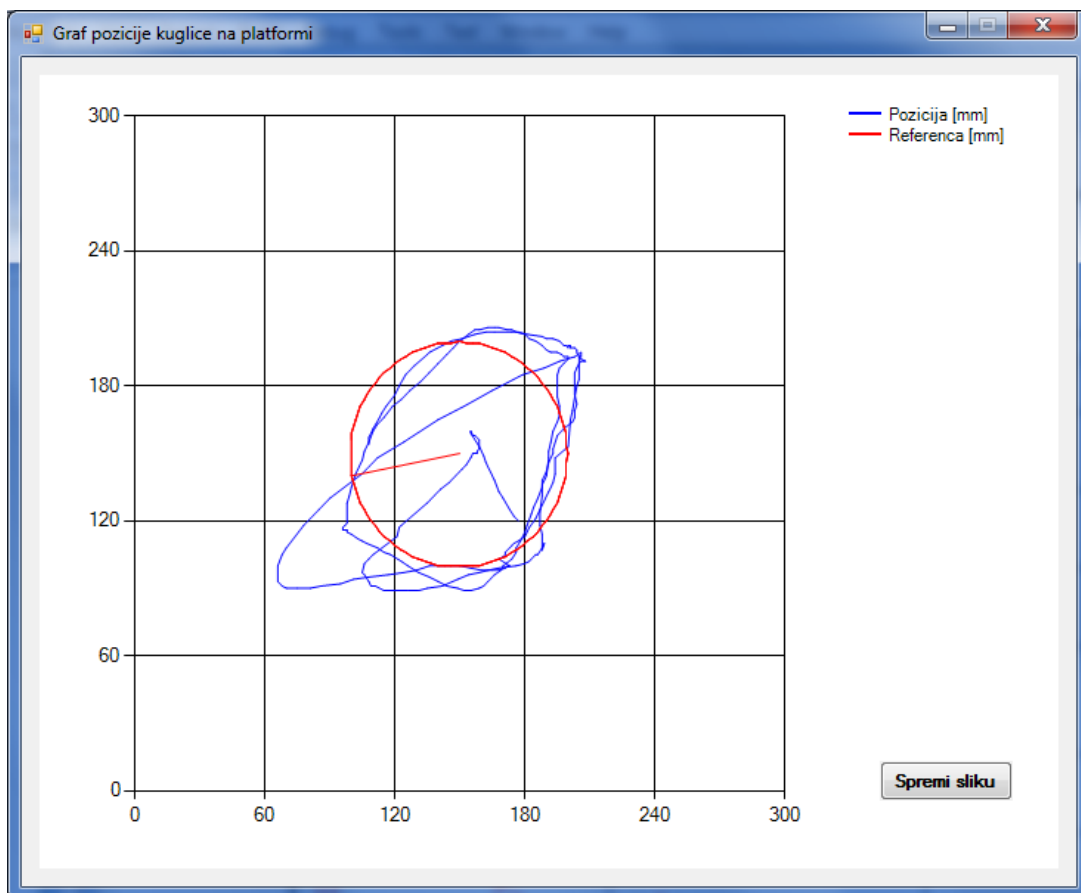
Slika 3.40: Kartica glavne aplikacije za podešavanje postavki osnovnog PD regulatora i referentnog signala

3.5.1.5. Iscrtavanje odziva

Iscrtavanje odziva sustava implementirano je u glavnoj aplikaciji korištenjem Microsoft Chart Controlsa. Glavna aplikacija ima mogućnost iscrtavanja tri tipa odziva: pozicije loptice na platformi, pozicije loptice na platformi za svaku os u odnosu na vrijeme te upravljačkih signala za svaku os u odnosu na vrijeme. Realizirana je mogućnost spremanja slika na disk osobnog računala u *.png* ili *.jpg* formatu.

Pri svakom koraku diskretizacije upravljački dio aplikacije spremi podatke o vremenu, poziciji loptice na platformi, referentnim položajima loptice te upravljačkim signalima u odgovarajuću *log* datoteku. Krajem upravljačkog eksperimenta *log* datoteke se zatvaraju za pisanje te postaju izvor podataka za funkcije iscrtavanja odziva sustava.

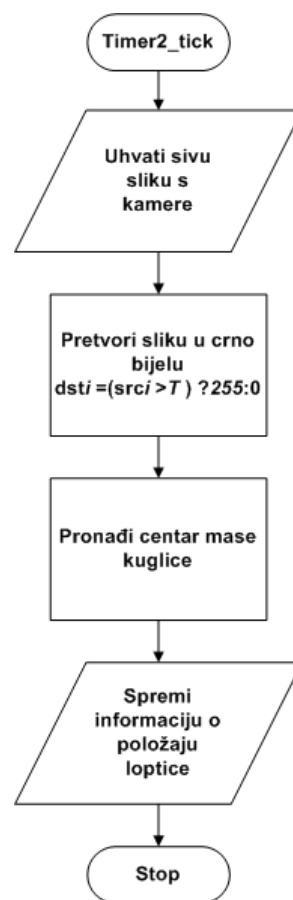
Pozivom za iscrtavanje odziva sustava pokreće se funkcija `dohvati()`. Funkcija čita sve podatke spremljene unutar *log* datoteka. Ukoliko je čitanje podataka bilo uspješno relevantni podatci prebacuju se u liste *lista_vrijeme*, *x_list* i *y_list*. Poznavajući broj podataka koji će se iscrtati poziva se funkcija `konfiguriraj_graf()`. Funkcija postavlja parametara grafa koji će se iscrtati (raspon osi, boje pojedinih signala i sl.). Nakon što su postavljeni parametri grafa koji će se iscrtavati poziva se funkcija `crtaj2()` koja za sve vrijednosti varijabli spremljenih u liste odgovarajućim naredbama dodaje točke na graf. Izgled grafa odziva pozicije loptice na platformi dan je slikom 3.41.



Slika 3.41: Isctavanje odziva pozicije loptice na platformi

3.5.2. Algoritam obrade slike

Algoritam obrade slike napisan je u C# programskom jeziku te su korištene već gotove biblioteke OpenCV-a. OpenCV (engl. Open Source Computer Vision Library) je skup za javnost otvorenih biblioteka za programe kojima se obavlja analiza slike ili videa u realnom vremenu. OpenCV sadrži čitav niz razvijenih algoritama, od onih u čestoj uporabi (crtanje geometrijskih likova i pisanje po slici, pretvorba između više formata slike i raznih filtara – računanje gradijenta, rubova, usrednjavanje slike medijan filtrom itd.) do specifičnih algoritama (proračun optičkog toka slike, računanje histograma, momenata, Houghove transformacije, Kalmanovih filtara itd.). OpenCV izvorno je razvijen za C++ programski jezik. Kako je glavna aplikacija razvijena u programskom jeziku C# korišten je Emgu CV *wrapper* koji omogućava istu funkcionalnost kao i izvorni OpenCV ali u programskom jeziku C#. Dijagram toka algoritma obrade slike dan je slikom 3.42.



Slika 3.42: Dijagram toka algoritma obrade slike

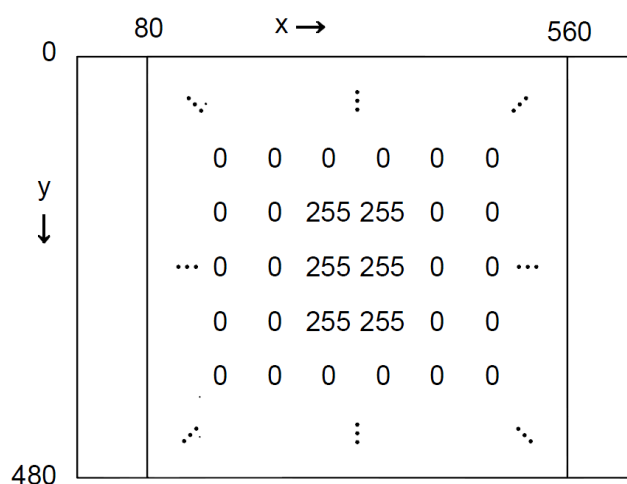
Ukoliko je algoritam obrade slike programski uključen aktivira se *Timer2* kontrola. Unutar postavki *Timer2* kontrole postavljen je period od 0.05 s pošto taj period označava brzinu obrade slika kamere od 20 *fps*. Ukoliko je *Timer* kontrola aktivna ona generira *događaj Timer2_tick* (eng. event) koji se izvodi svako unaprijed postavljenog perioda. Algoritam obrade slike izvodi se dakle na svaki *tick Timera*. Ulaskom u funkciju obrade slike inicira se hvatanje sive slike s kamere. Siva slika sprema se odgovarajući `Image<Gray, byte>` razred koji u Emgu CV predstavlja slike nad kojima se izvode različite manipulacije. Unutar navedenog razreda sama slika predstavljena je poljem veličine

640 x 480. Svaka vrijednost polja nalazi se u rasponu od 0 do 255 ovisno o osvjetljenosti pojedine točke slike.

Kako bi se pronašla pozicija loptice na platformi, izvodi se segmentacija na temelju praga (*eng.* thresholding). Kako je platforma crne boje a loptica je sjajne površine za očekivati je da će segmentacija na temelju praga odvojiti sliku objekta tj. loptice od pozadine tj. platforme. Segmentacija se izvodi provjerom svih točaka slike. Ukoliko je vrijednost točke slike veća od vrijednosti praga (*eng.* thresh) dodjeljuje joj se vrijednost 255 što predstavlja bijelu boju, suprotno dodjeljuje joj se vrijednost 0 što predstavlja crnu boju. Korisnik može mijenjati vrijednost praga unutar glavne aplikacije. Ovisno o vanjskim uvjetima osvjetljenja različiti izbor vrijednosti praga daje različite rezultate.

Dobivena crno bijela slika koristi se u algoritmu za pronalaženje objekta tj. loptice na platformi. Algoritam ne pretražuje cijelu sliku već samo njen predodređen dio. Pošto je platforma kvadratnog oblika a slika dobivena s kamere pravokutnog određeno je da algoritam pretražuje centralni kvadratni dio slike dobivene s kamere. Algoritam izvodi pretraživanje nad područjem shematski prikazanim slikom 3.43.

Pronalaženje objekta tj. loptice na platformi svodi se na pronalaženje centra mase objekta. Binarna crno bijela slika ima oblik prikazan slikom 3.43. Područje prikazano bijelom bojom (tj. brojem 255 na slici) definira se kao područje R koje predstavlja objekt.



Slika 3.43: Prikaz crno-bijele slike dobivene segmentacijom

Područje R se može opisati nizom brojeva m_{kj} koji karakteriziraju oblik područja a nazivaju se momenti od R . Momenti se definiraju na sljedeći način:

$$m_{kj} = \sum_{(x,y) \in R} x^k y^j \quad (3-54)$$

Gdje je x stupac točke s težinom 255 u području R a y redak točke s težinom 255 u području R . Za centar mase tj. sredinu objekta tada vrijedi:

$$\begin{aligned}
 A &= m_{00} \\
 x_c &= \frac{m_{10}}{m_{00}} \\
 y_c &= \frac{m_{01}}{m_{00}}
 \end{aligned}
 \tag{3-55}$$

Dobivene su koordinate centra loptice na platformi izraženi kao točka na slici dobivenoj s kamere tj. u pikselima. Pošto se pozicija loptice s kamere koristi kao povratna veza sustava u kojem je sve izraženo *SI* sustavom tj. u metrima potrebno je normirati vrijednosti loptice iz piksela u metre. Uzevši u obzir veličinu platforme ($0.3 \times 0.3 \text{ m}$) te područje na kojem se na slici nalazi platforma $(0,0), (560,480)$ izvodi se normiranje idućom linearnom relacijom:

$$\begin{aligned}
 x_{mjereni} &= 0.625x_c - 50.625 \\
 y_{mjereni} &= 0.625y_c
 \end{aligned}
 \tag{3-56}$$

Normirane vrijednosti spremaju se u odgovarajuće varijable te ih upravljački dio koristi kao signal povratne veze. Upravljački dio uvijek dobiva najsvježiji podatak o trenutnoj poziciji loptice pošto se algoritam obrade slike izvodi sa dva puta bržim vremenom diskretizacije od algoritma upravljanja.

Prikazani algoritam, iako jednostavan, za ovu primjenu pokazao se odličnim. Temeljni zahtjev na algoritam obrade slike primjenjiv u ovakvom sustavu automatskog upravljanja je da se izvodi u stvarnom vremenu. Zbog jednostavne izvedbe, algoritam za obradu slike nije procesorski zahtjevan te se postigla brzina obrade jedne slike od otprilike 0.05 s što znači da se u jednoj sekundi uspije obraditi 20 slika s kamere tj. 20 fps . Ovakav rezultat garantira uvijek svjež podatak za upravljački dio sustava.

Konfiguracija slike koju prima kamera (crna pozadina platforme i svijetla loptica), uz korišteni algoritam, uvjetuje gotovo stopostotnu točnost i preciznost algoritma za obradu slike. Robusnost algoritma jedino može biti narušena promjenjivim vanjskim osvjetljenjem ali se vanjski utjecaj može smanjiti korisničkom intervencijom na vrijednost praga. Slikom 3.44 dan je prikaz pronalazjenja kuglice na crno-bijeloj segmentiranoj slici dobivenoj s kamere.



Slika 3.44: Pronalazak kuglice na platformi

3.5.3. Programiranje korisničkog algoritma upravljanja

Najvažnija funkcija razvijene programske podrške je mogućnost dodavanja novih algoritama upravljanja u samoj aplikaciji bez potrebe korištenja alata za razvoj programske podrške. Korisnički upravljački algoritam dinamički se prevodi u stvarnom vremenu (*eng.* Dynamic Compilation). Korisnik unosi novi upravljački algoritam u *textbox* koji se nalazi u samoj aplikaciji te ima mogućnost učitavanja ili spremanja napisanih upravljačkih funkcija. Korisnik se pri izradi novog upravljačkog algoritma mora pridržavati unaprijed određene strukture funkcije koja implementira razvijeni algoritam ali unutar funkcije ima potpunu slobodu pisanja vlastitog koda. Potrebna struktura korisničke funkcije dana je programskim odsječkom u nastavku.

```
using System;
public class Regulator
{
    static double varijabla;

    public double [] algoritam(double x_error, double y_error, double
x_mjereni, double y_mjereni)
    {
        //korisnicki kod

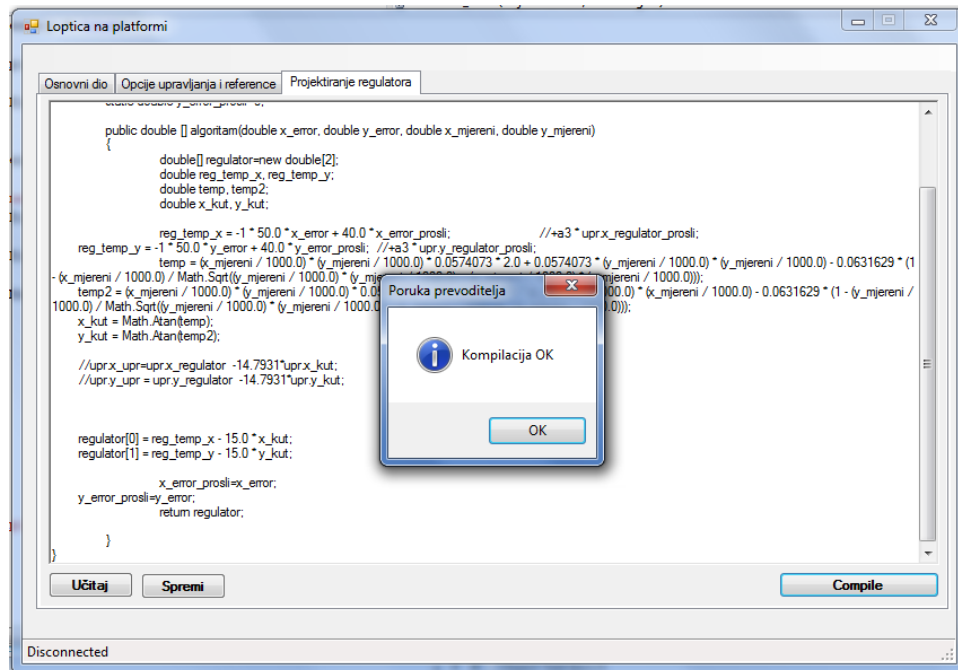
        //Korisnik mora glavnoj aplikaciji vratiti vrijednost u polju
regulator

        regulator[0] = x_vrijednost;
        regulator[1] = y_vrijednost;

        return regulator;
    }
}
```

Dinamičko prevođenje izvodi se metodama dostupnim u sklopu .NET Framework biblioteka. Korisnički dio koda prevede se (*eng.* compile) te se spremi u radnu memoriju računala koristeći razred `CompilerResults`. Pokretanjem upravljačkog dijela glavne aplikacije iz navedenog razreda kreira se razred `Assembly`. Iz razreda koji predstavlja strojni jezik stvara se razred koji predstavlja korisnički unesen komad koda. Razred sadrži metodu `algoritam` koja predstavlja funkciju koju je korisnik napisao i koja predstavlja upravljački algoritam sustava. Funkcija tj. metoda koja sadrži upravljački algoritam poziva se pri svakom koraku diskretizacije upravljačkog dijela. Kao ulaze prima vrijednosti regulacijskog odstupanja za obje osi i mjerenu poziciju loptice na platformi. Kao izlaz daje vrijednosti upravljačkih signala za obje osi koje se dalje koriste pri upravljanju sustavom. Ukoliko korisnik učini bilo kakvu grešku u kodu pri prevođenju glavna aplikacija će javiti neuspješnost prevođenja te mjesto

učinjene greške (linija koda greške). Izgled kartice korisničkog sučelja za unos novog algoritma upravljanja prikazan je slikom 3.45.



Slika 3.45: Kartica glavne aplikacije za korisničko unošenje algoritma upravljanja

Dinamičko prevođenje moguće je izvoditi u stvarnom vremenu čak i dok se upravlja sustavom. Ova činjenica olakšava fino podešavanje parametara upravljačkog algoritma što se pri projektiranju sustava automatskog upravljanja redovito radi. Svojstvo dinamičkog prevođenja glavne aplikacije čini je samostalnim i neovisnim rješenjem. Uz ostale funkcionalnosti glavna aplikacija primjenjiva je kao jedini alat projektiranja upravljačkog algoritma za promatrani sustav loptice na platformi.

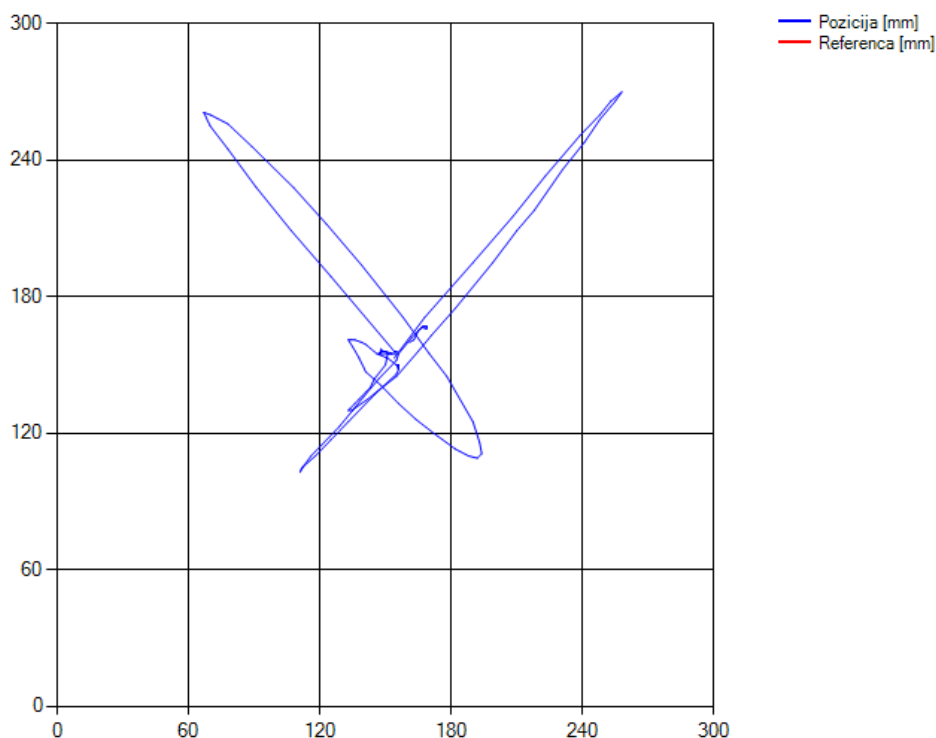
4. Rezultati i rasprava

Funkcionalnost sustava ispitana je izvođenjem različitih eksperimenata na realnom sustavu loptice na platformi. Kao upravljački algoritam korišten je modificirani projektirani pokazni regulator s pomoćnom upravljačkom petljom po kutu zakrivljenosti platforme. Pokazni regulator implementiran je jednadžbama diferencija koristeći mogućnost unosa vlastitog algoritma upravljanja u razvijenoj glavnoj aplikaciji. Parametri regulatora dobiveni sintezom i simulacijama u Matlab Simulink okruženju nisu se pokazali dobrima te su parametri regulatora ugađani različitim eksperimentima na stvarnom sustavu. Korišteni pokazni algoritam upravljanja u obliku korisnički unesene funkcije dan je u prilogu D.

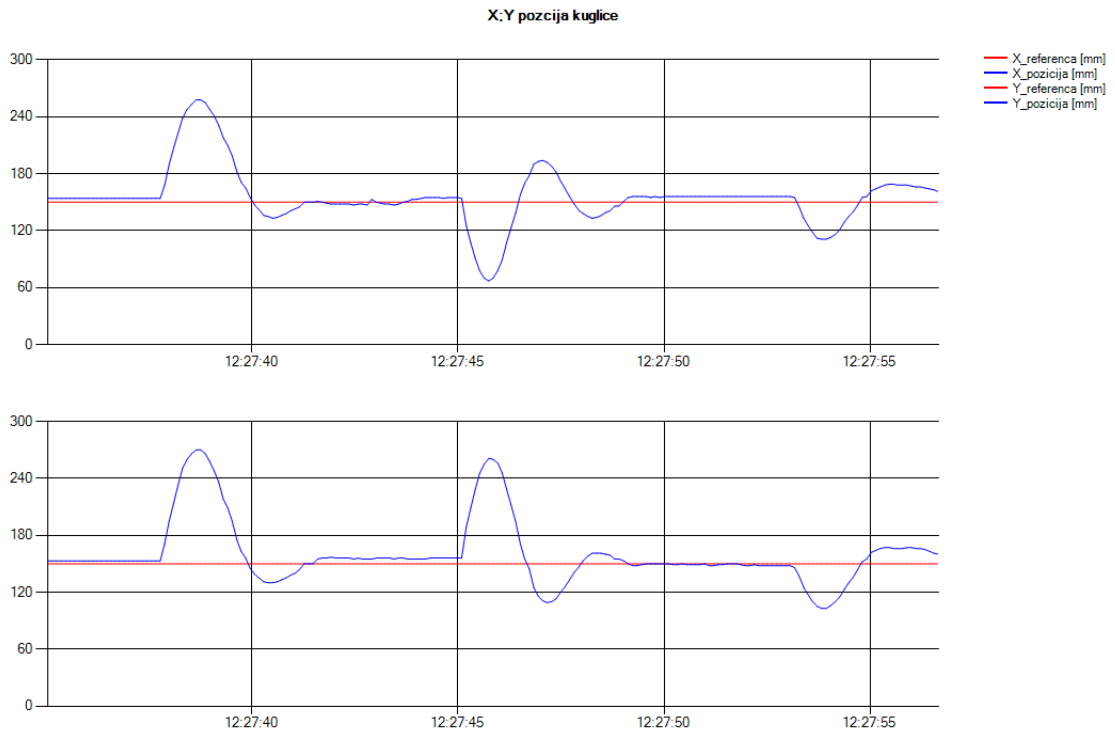
Izvedeni su različiti eksperimenti upravljanja s različitim zadanim referentnim vrijednostima. Eksperimenti su popraćeni odzivima varijabli sustava dobivenih iscrtavanjem u glavnoj aplikaciji.

4.1. Eksperiment stabilizacije loptice u zadanoj točki

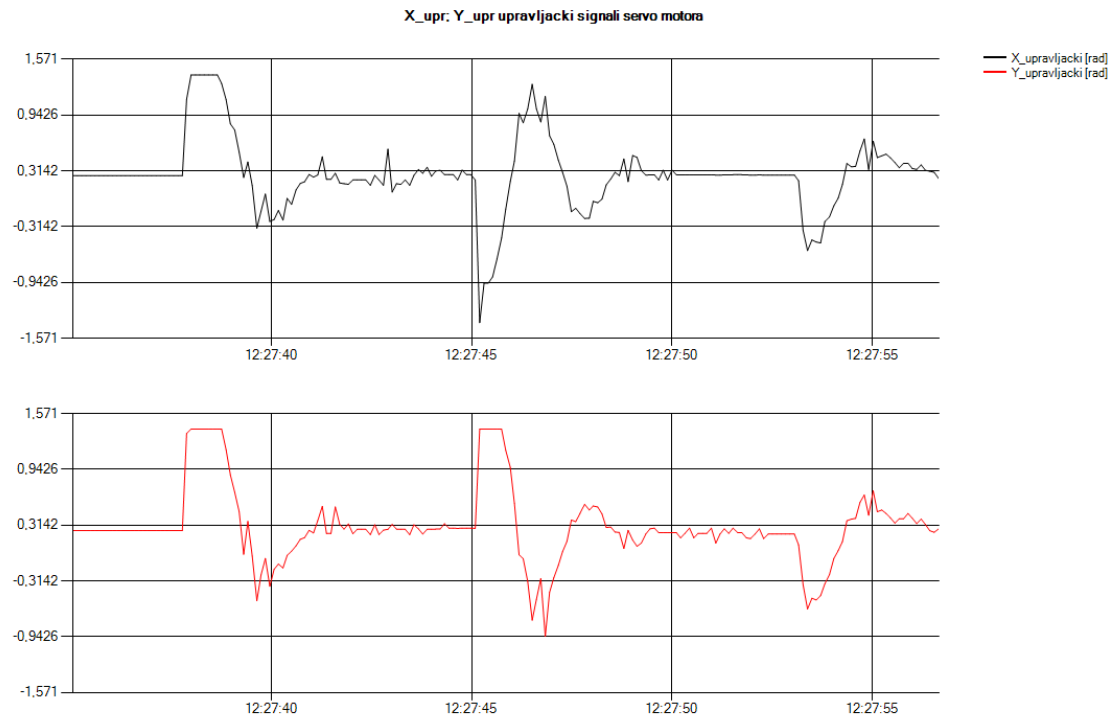
Pri eksperimentu stabilizacije loptice kao referentni položaj loptice na platformi korištena je točka s vrijednošću $(0.15\text{ m}, 0.15\text{ m})$ što predstavlja sredinu platforme. Pri eksperimentu ispitivano je vladanje sustava na vanjsku poremećajnu veličinu koja se manifestirala kao guranje loptice iz točke u kojoj se nalazila. Na slikama 4.1, 4.2 i 4.3 prikazani su dobiveni odzivi sustava (pozicije loptice i upravljački signali). Na odzivima se vidi da algoritam upravljanja uspješno stabilizira lopticu u željenoj centralnoj točki platforme. Pri djelovanju vanjske poremećajne veličine algoritam uspješno vraća lopticu u željenu točku. Odziv loptice pokazuje nadvišenje koje je rezultat dinamike sustava i korištenog PD pokaznog regulatora.



Slika 4.1: Pozicija loptice na platformi



Slika 4.2: Pozicija loptice po x i y osi

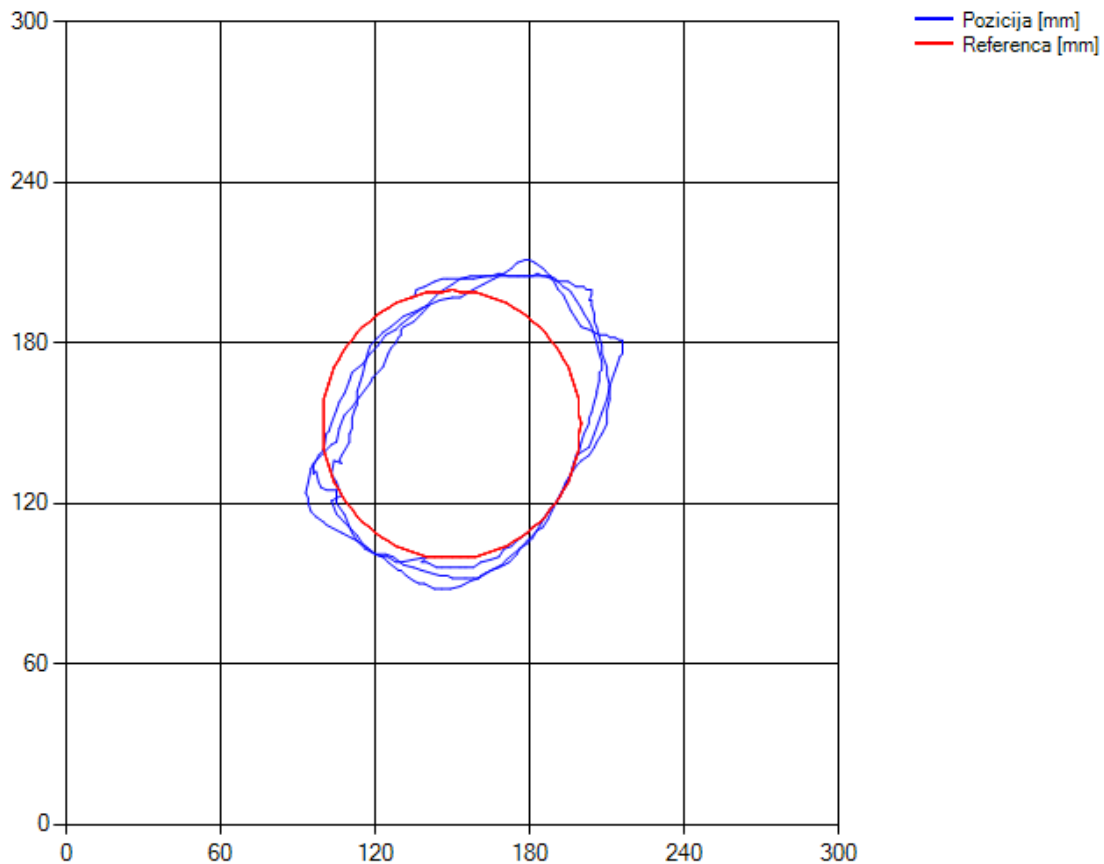


Slika 4.3: Upravljački signali servomotora

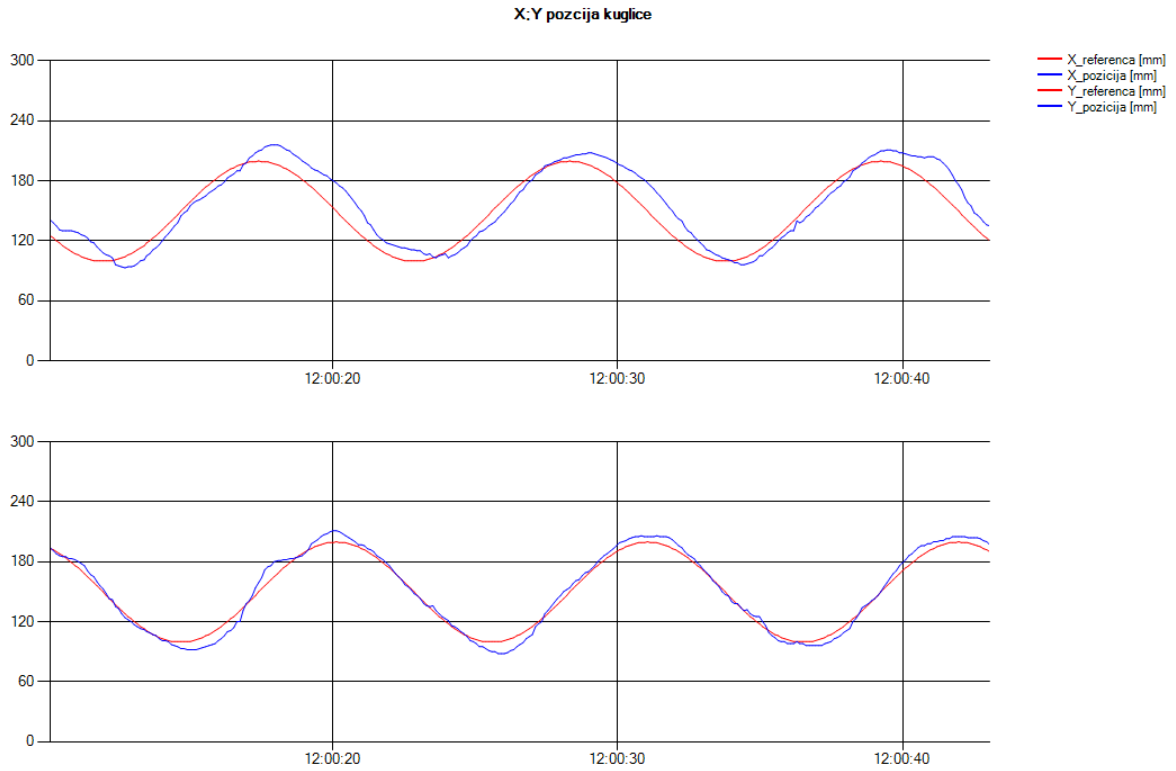
4.2. Eksperiment kružnice

Pri eksperimentu kružnice kao referentni signal odabrana je sinusna pobuda oblika $r_x(t) = 0.05 \cdot \sin\left(\frac{2\pi}{10}t + \frac{\pi}{2}\right) + 0.15$ za x os odnosno $r_y(t) = 0.05 \cdot \sin\left(\frac{2\pi}{10}t\right) + 0.15$ za y os. Ta dva signala predstavljaju kružnicu s centrom u $(0.15\text{ m}, 0.15\text{ m})$ te radijusom 0.05 m .

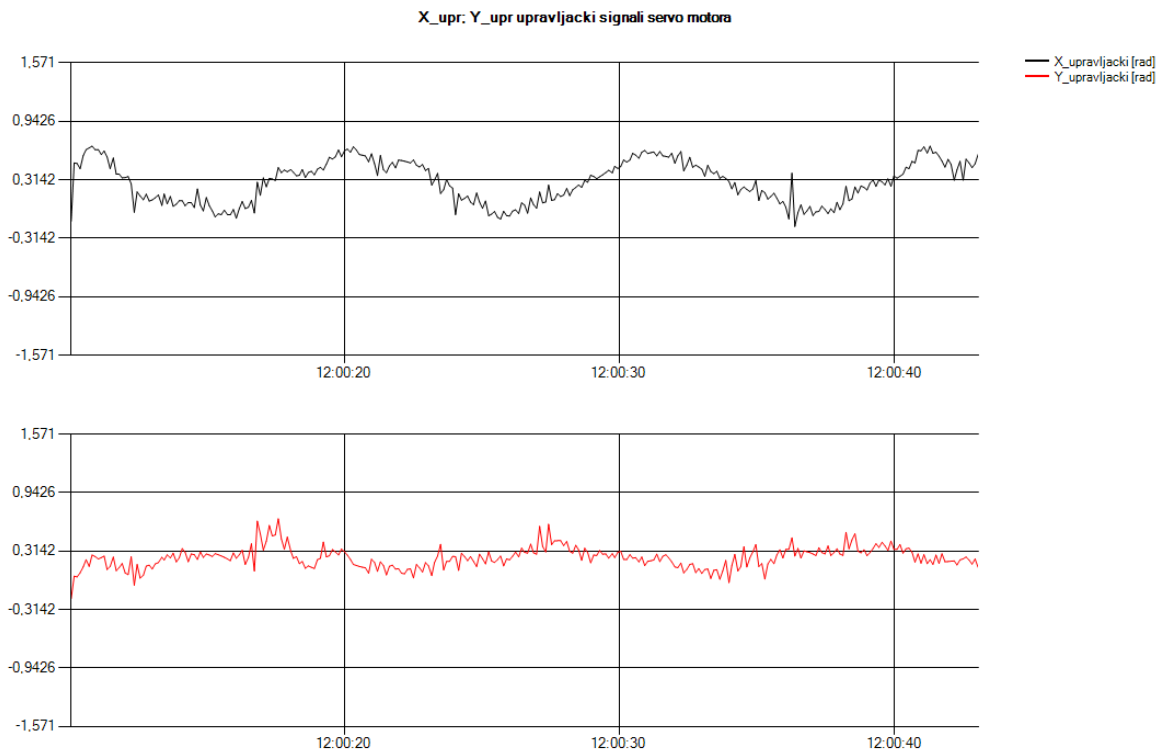
Na slikama 4.4, 4.5 i 4.6 prikazani su dobiveni odzivi sustava (pozicije loptice i upravljački signali). Iz odziva sustava vidi se da loptica prati željenu referentnu poziciju ali uz određeno regulacijsko odstupanje i brzinsku pogrešku koja je specifična za korišteni pokazni PD regulator. Regulacijsko odstupanje uzrokovano je zakrivljenošću platforme te sitnim oštećenjima površine platforme koja stvaraju problem pri upravljanju. U ovom eksperimentu korištena je pomoćna upravljačka petlja po kutu zakrivljenosti platforme što značajno poboljšava kvalitetu upravljanja.



Slika 4.4: Pozicija loptice na platformi



Slika 4.5: Pozicija loptice po x i y osi

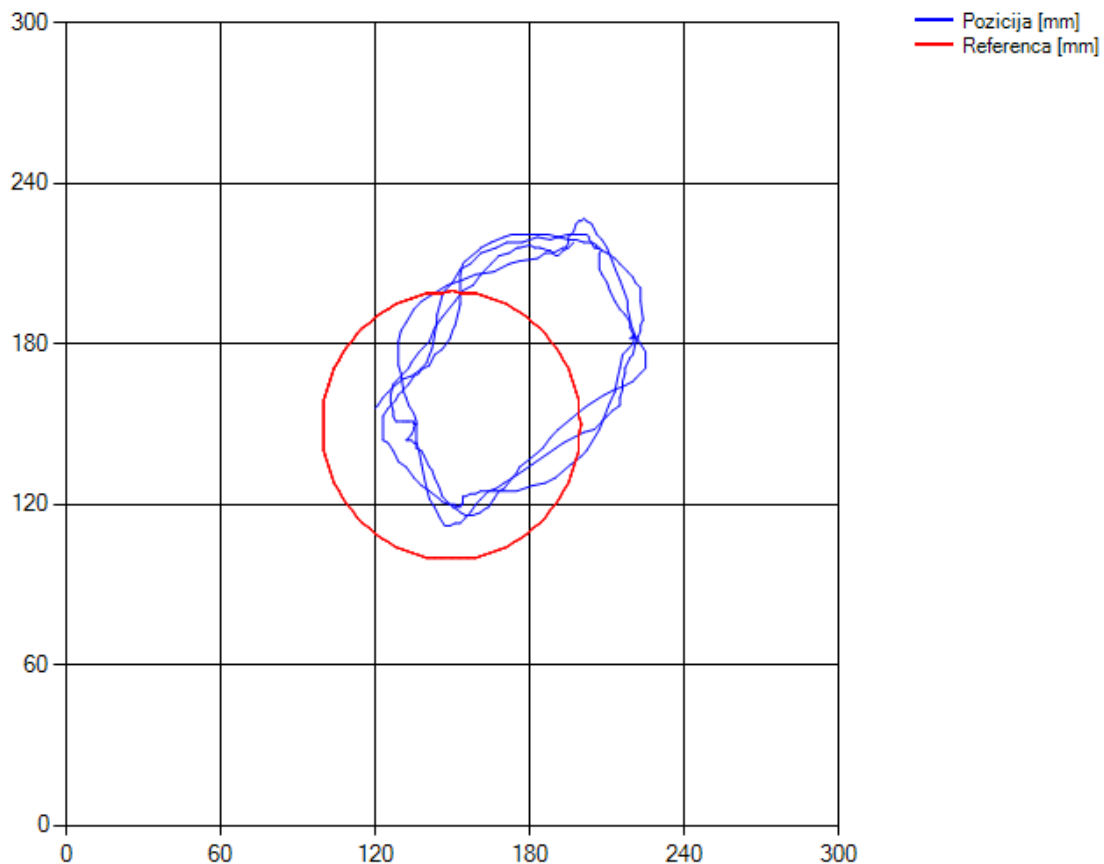


Slika 4.6: Upravljački signali servomotora

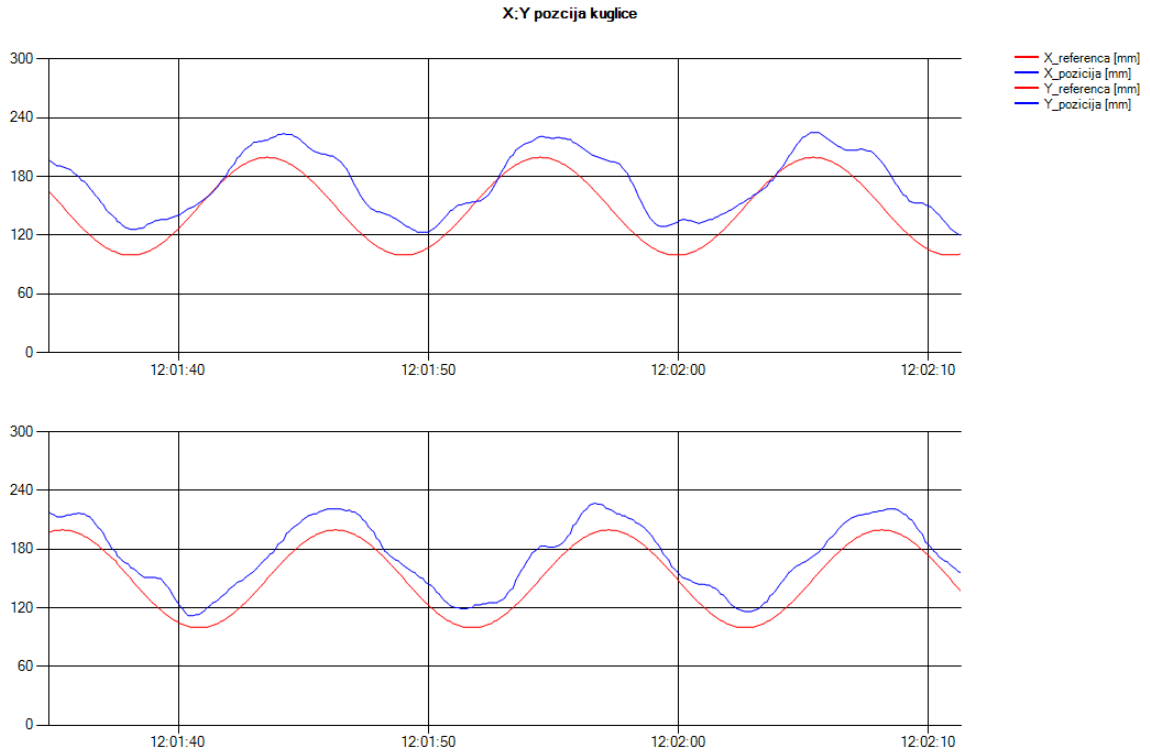
4.3. Eksperiment kružnice bez pomoćne upravljačke petlje

Ovaj eksperiment izveden je pod istim uvjetima i s istim signalom reference kao i eksperiment izveden u poglavlju 4.3. Upravljački algoritam nije sadržavao pomoćnu upravljačku petlju koja upravlja kutom zakreta servomotora po izračunatom kutu zakrivljenosti plohe platforme.

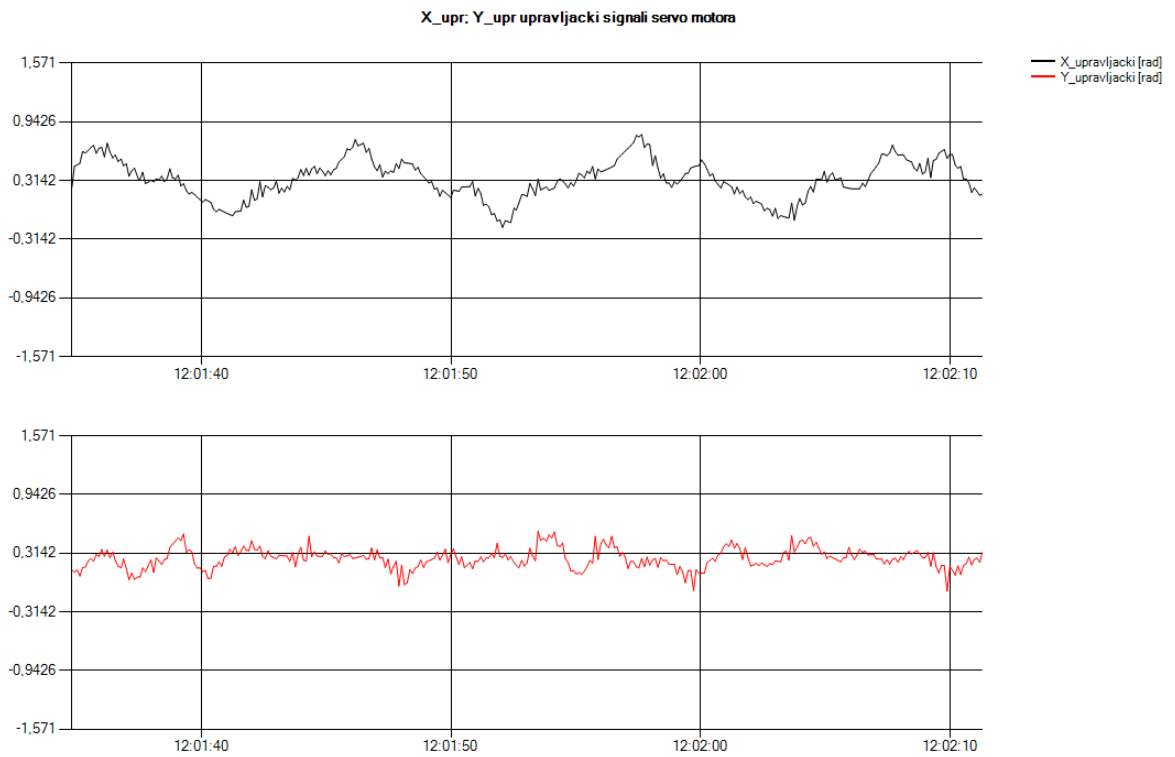
Odzivi sustava dani su slikama 4.7, 4.8 i 4.9. Iz danih odziva se vidi da je utjecaj zakrivljenosti platforme na kvalitetu upravljanja sustava velik. Platforma je savinuta prema slobodnom vrhu platforme te je kružnica koju opisuje loptica sukladno tome pomaknuta prema slobodnom savinutom vrhu platforme.



Slika 4.7: Pozicija loptice na platformi



Slika 4.8: Pozicija loptice po x i y osi

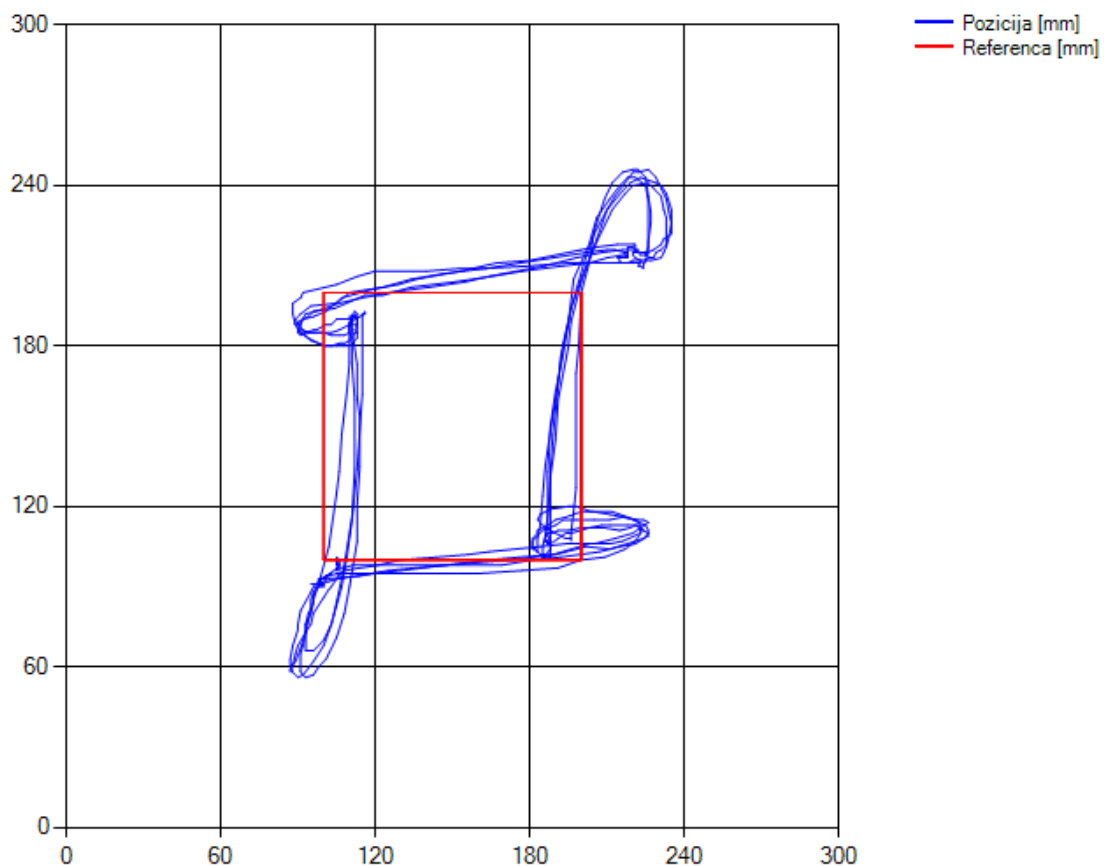


Slika 4.9: Upravljački signali servomotora

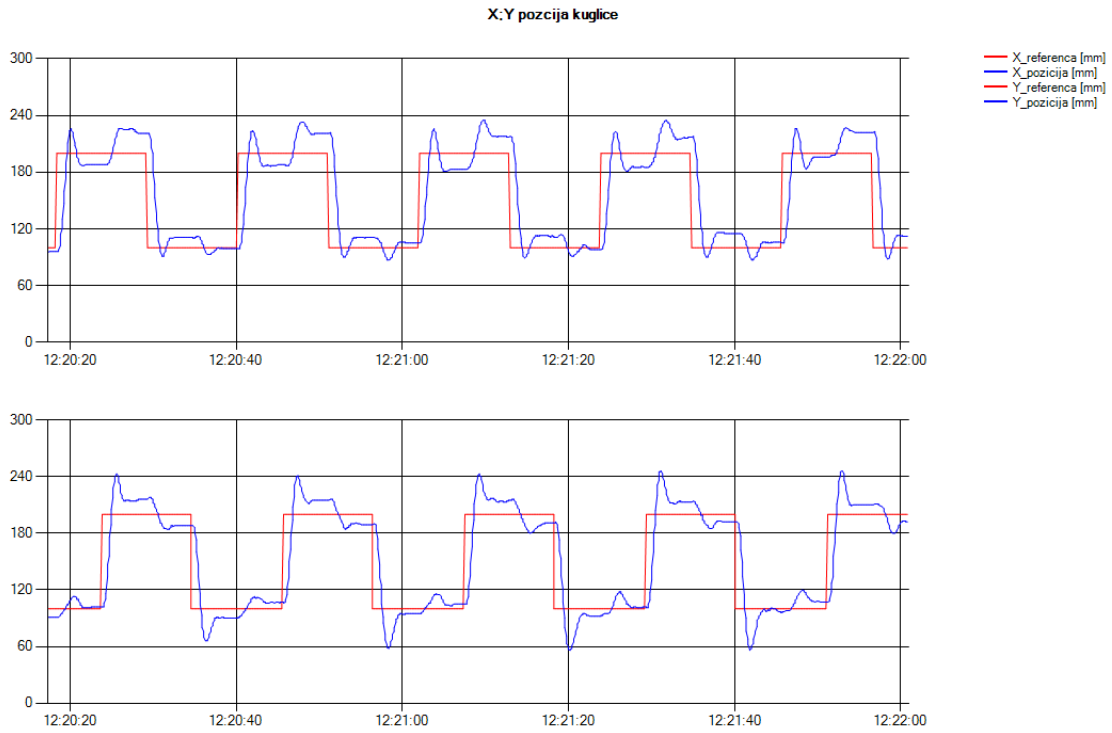
4.4. Eksperiment kvadrata

Eksperiment je izveden uz korištenje signala reference koji opisuje kvadrat s vrhovima u $(0.1\text{ m}, 0.1\text{ m})$ $(0.2\text{ m}, 0.1\text{ m})$ $(0.2\text{ m}, 0.2\text{ m})$ $(0.1\text{ m}, 0.2\text{ m})$. Korišten je pokazni upravljački algoritam s pomoćnom upravljačkom petljom.

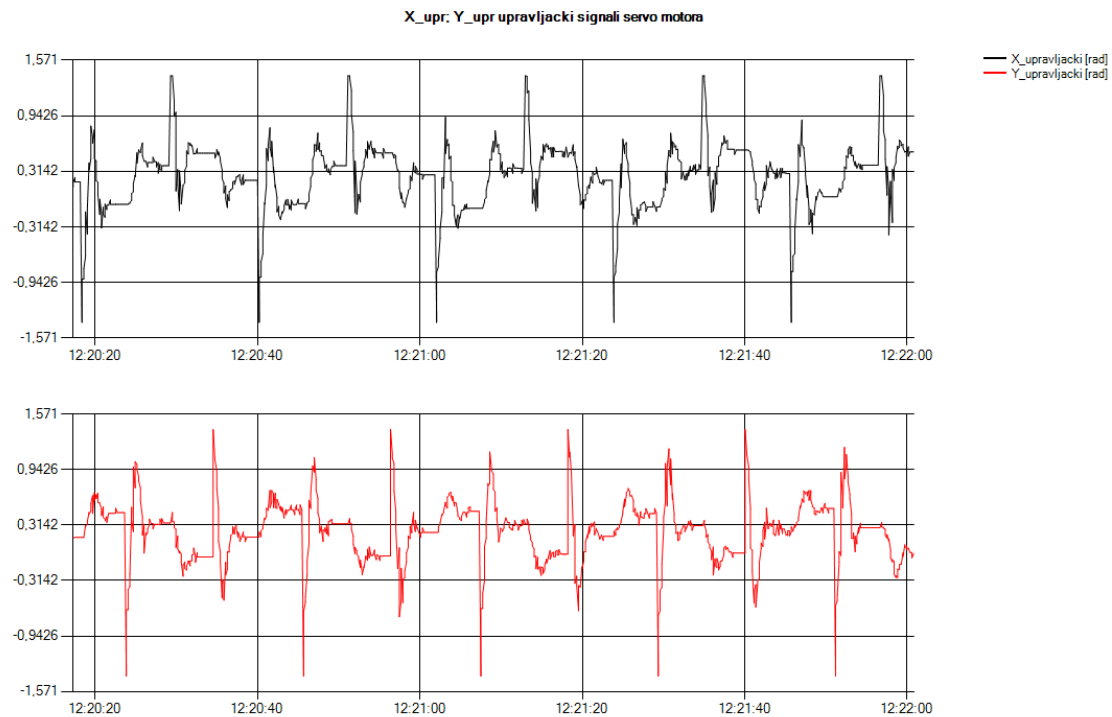
Odzivi sustava dani su slikama 4.10, 4.11 i 4.12. Iz danih odziva vidi se da optica dobro prati referentnu veličinu uz prisutno regulacijsko odstupanje te velika nadvišenja koja su rezultat korištenog pokaznog PD regulatora i velikog ubrzanja kojeg optica dobije pri promjeni referentne točke. Pomoćna upravljačka petlja po kutu zakrivljenosti plohe platforme znatno poboljšava kvalitetu upravljanja sustavom.



Slika 4.10: Pozicija optice na platformi



Slika 4.11: Pozicija loptice po x i y osi



Slika 4.12: Upravljački signali servomotora

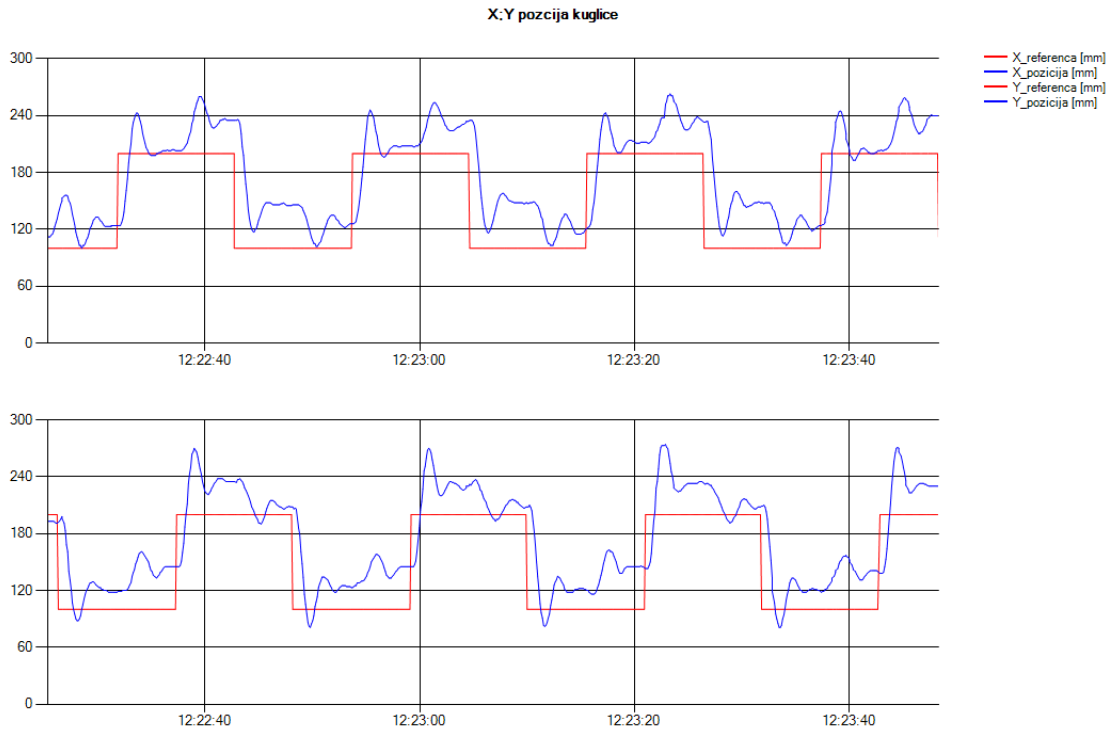
4.5. Eksperiment kvadrata bez pomoćne upravljačke petlje

Idući eksperiment izveden je pod istim uvjetima i s istim signalom reference kao i eksperiment izveden u poglavlju 4.4. Upravljački algoritam nije sadržavao pomoćnu upravljačku petlju koja upravlja kutom zakreta servomotora po izračunatom kutu zakrivljenosti plohe platforme.

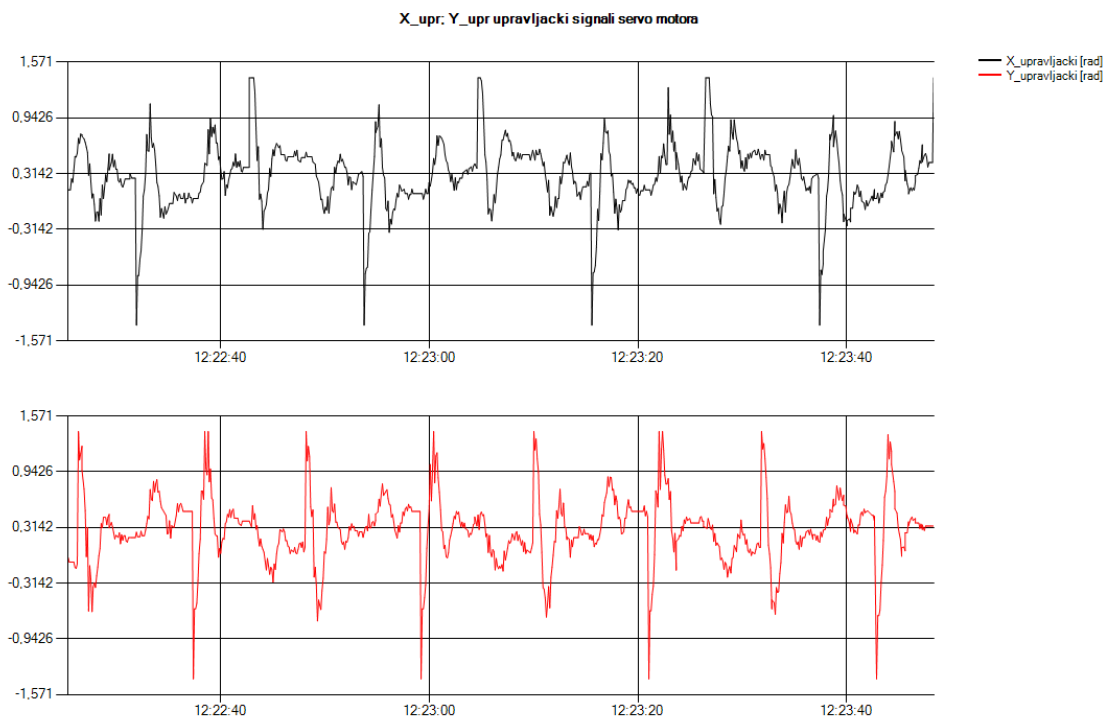
Odzivi sustava prikazani su na slikama 4.13, 4.14 i 4.15. Iz danih odziva se vidi da kvaliteta upravljanja značajno ovisi o korištenju pomoćne upravljačke petlje. Pošto se pomoćna upravljačka petlja ne koristi utjecaj zakrivljenosti platforme može se vidjeti na odzivima sustava. Kvadrat koji opisuje loptica pomaknut je prema slobodnom neučvršćenom vrhu platforme. Odziv pozicije pokazuje značajno regulacijsko odstupanje i nadvišenje.



Slika 4.13: Pozicija loptice na platformi



Slika 4.14: Pozicija loptice po x i y osi

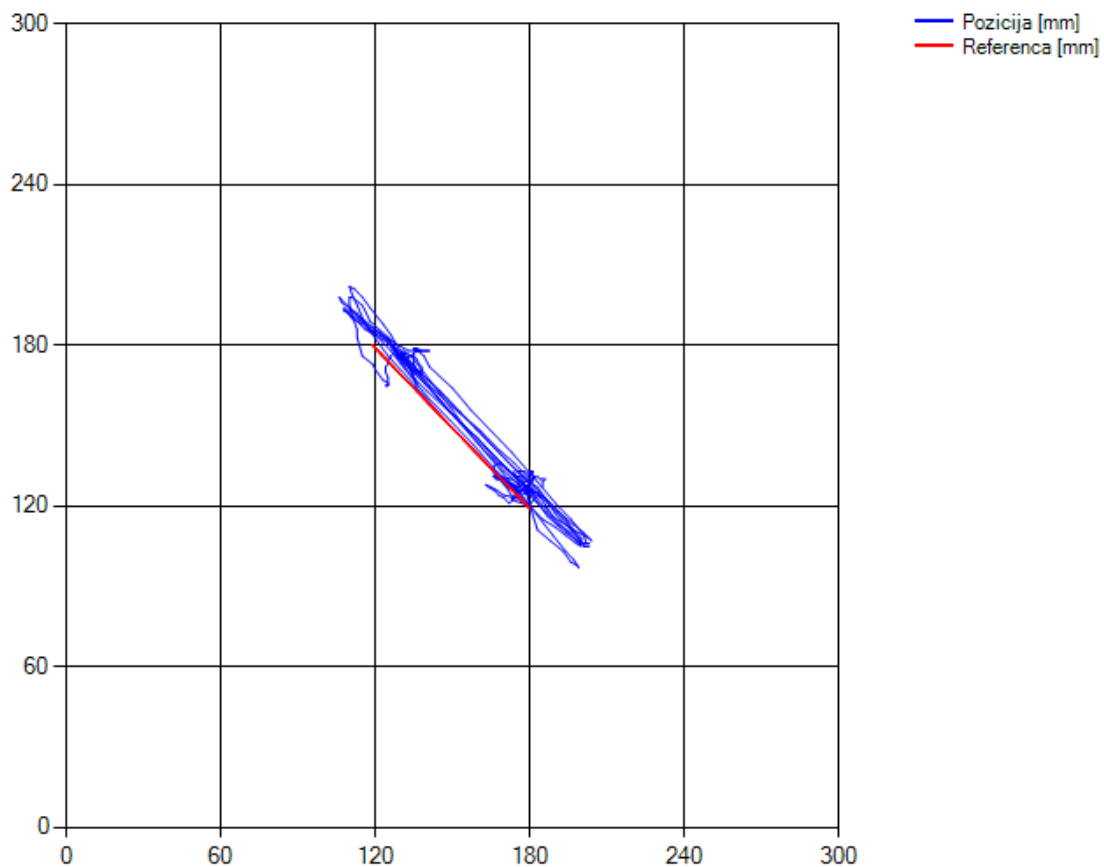


Slika 4.15: Upravljački signali servomotora

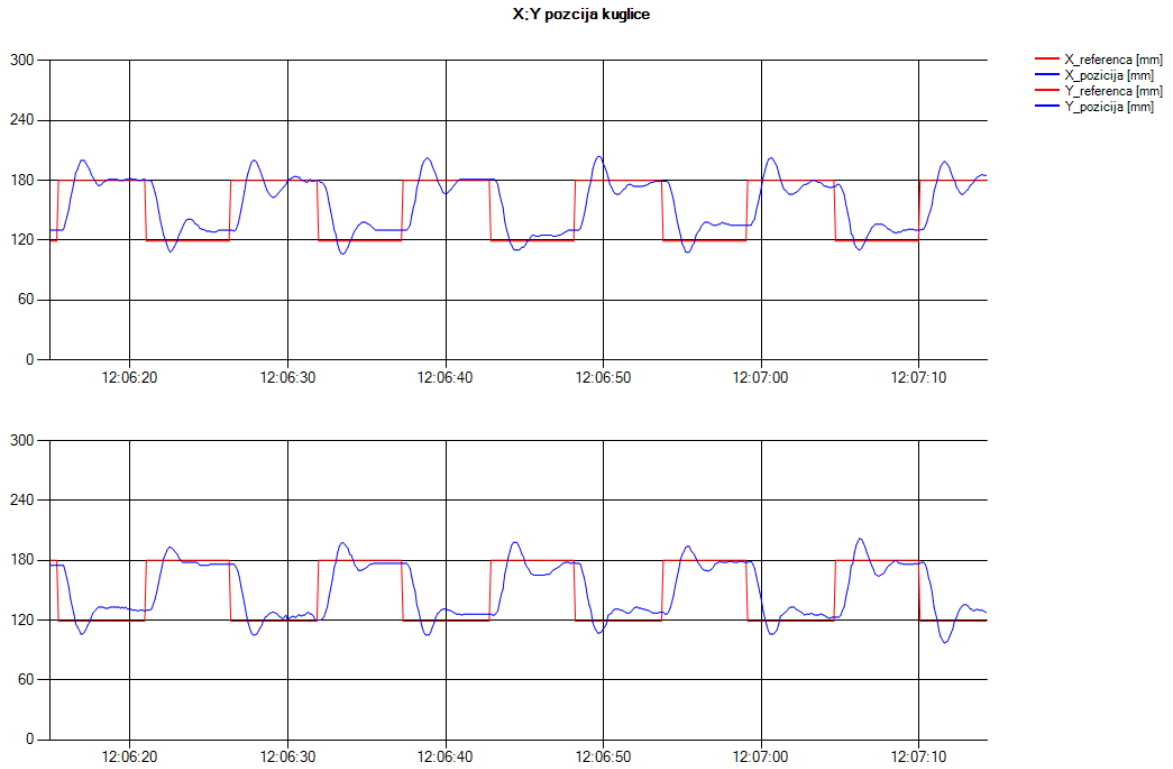
4.6. Eksperiment pravca

Pri eksperimentu pravca korišten je referentni signal koji opisuje dužinu s vrhovima u $(0.18\text{ m}, 0.12\text{ m})$ $(0.12\text{ m}, 0.18\text{ m})$. Korišten je pokazni upravljački algoritam s pomoćnom upravljačkom petljom.

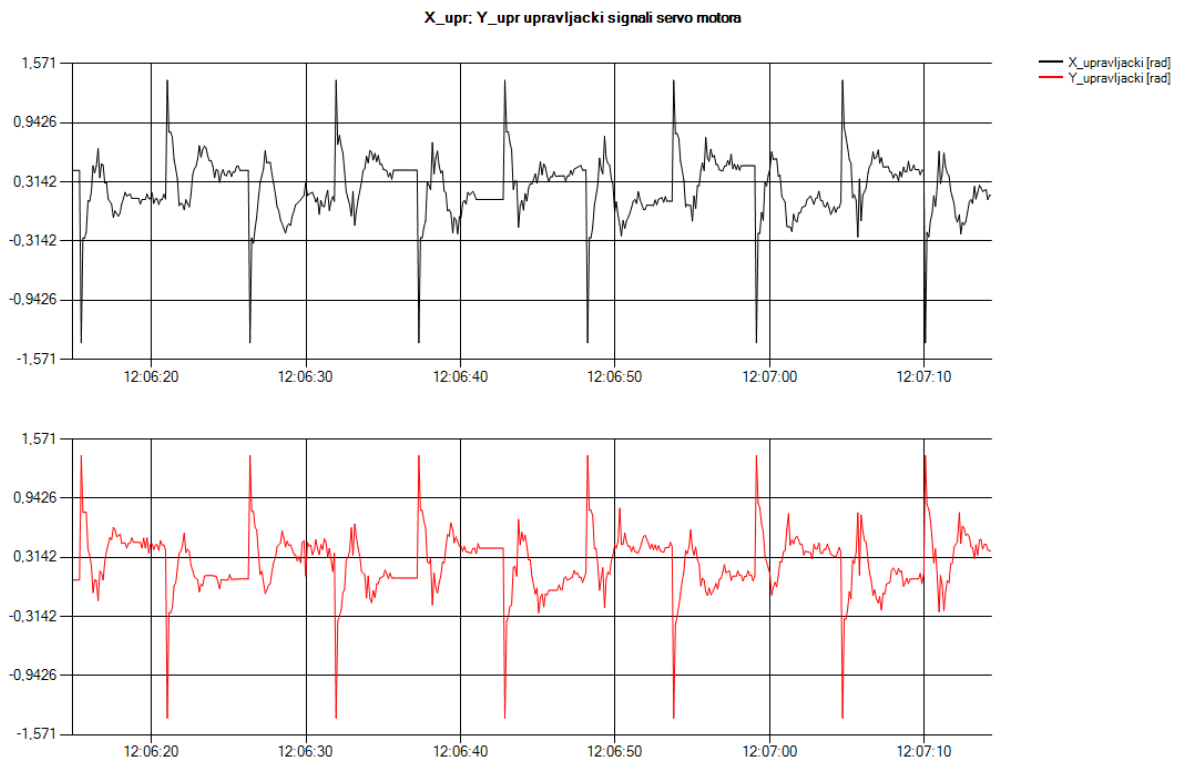
Odzivi sustava dani su slikama 4.16, 4.17 te 4.18. Iz danih odziva vidi se da loptica dobro prati referentni signal uz opet prisutno nadvišenje specifično za korišteni pokazni PD regulator. Regulacijsko odstupanje je neznatno. Kvaliteta upravljanja bitno je poboljšana korištenjem pomoćne upravljačke petlje.



Slika 4.16: Pozicija loptice na platformi



Slika 4.17: Pozicija loptice po x i y osi

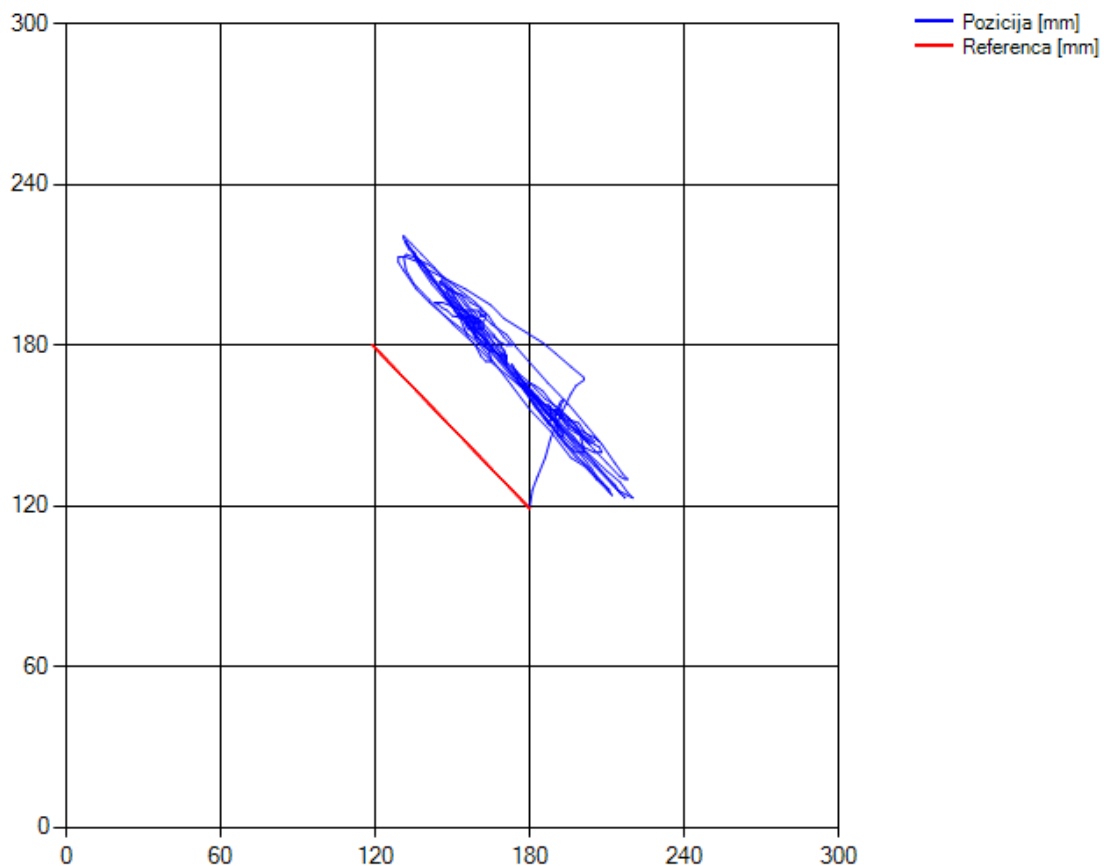


Slika 4.18: Upravljački signali servomotora

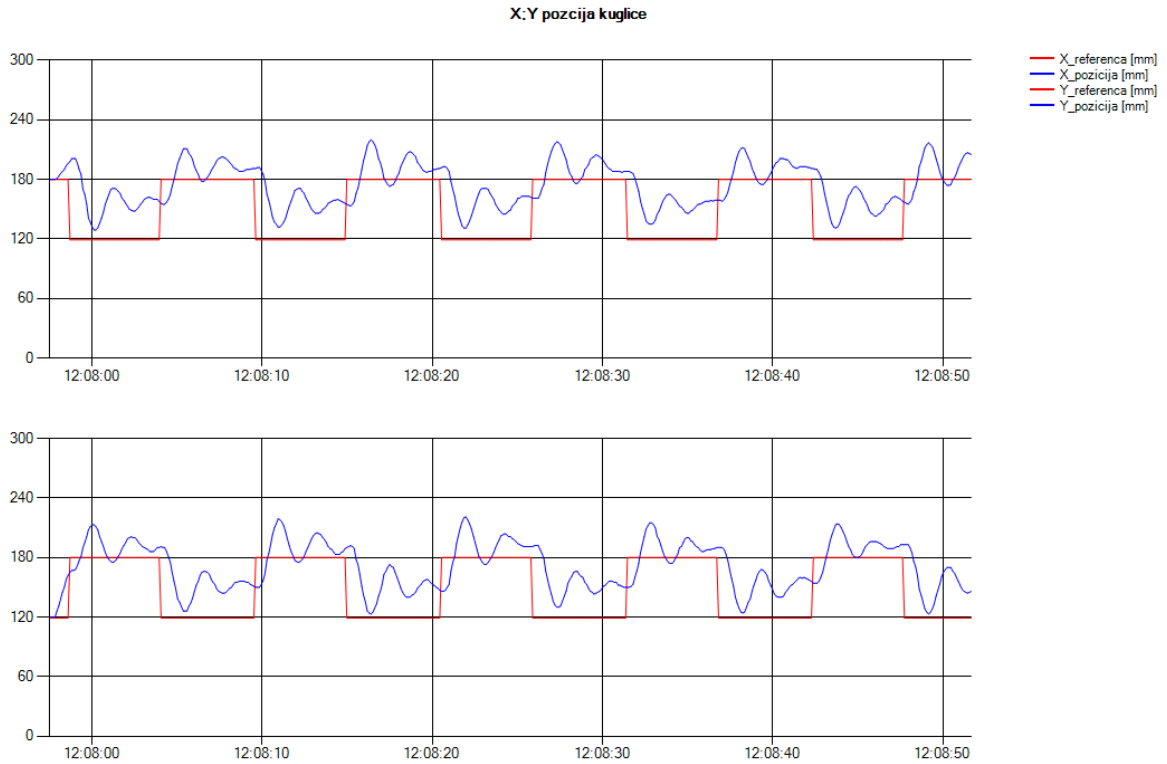
4.7. Eksperiment pravca bez pomoćne upravljačke petlje

Ovaj eksperiment izveden je pod istim uvjetima i s istim signalom reference kao i eksperiment izveden u poglavlju 4.4. Upravljački algoritam nije sadržavao pomoćnu upravljačku petlju koja upravlja kutom zakreta servomotora po izračunatom kutu zakrivljenosti plohe platforme.

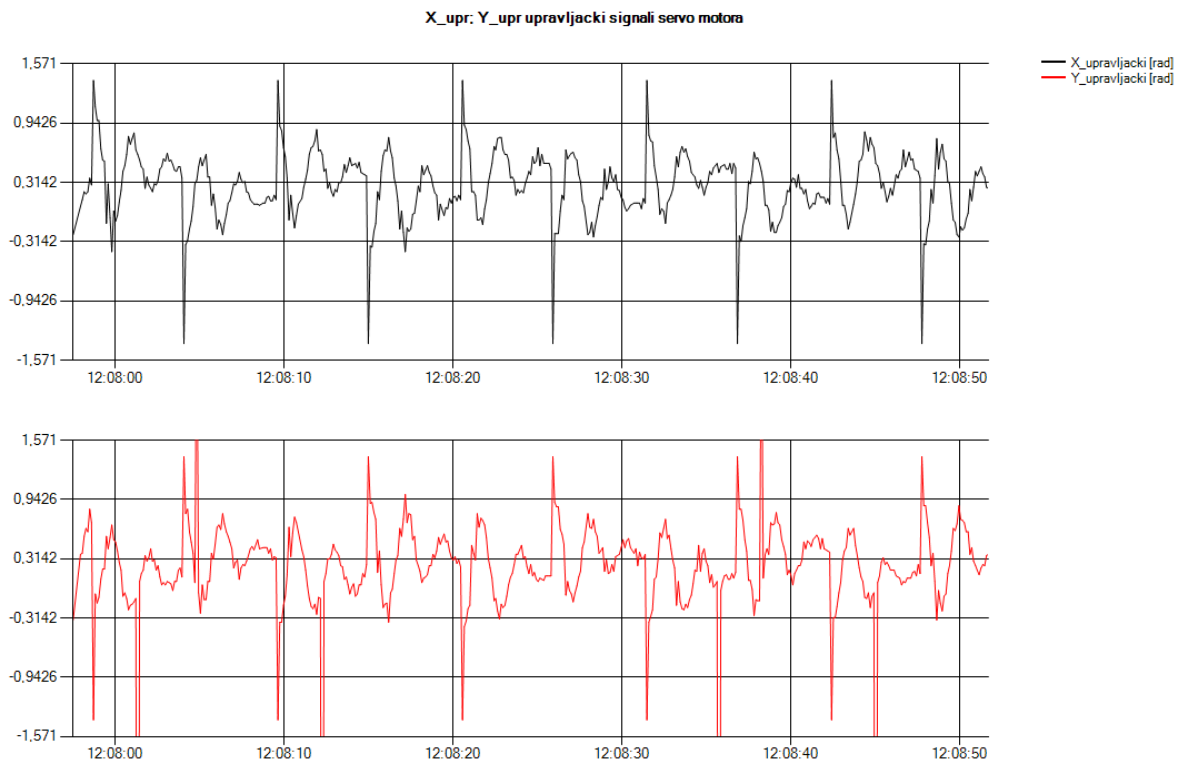
Odzivi sustava dani su slikama 4.19, 4.20 i 4.21. Kvalitete upravljanja bitno je pogoršana zbog utjecaja zakrivljenosti platforme. Prisutno je značajno regulacijsko odstupanje te je pravac koji čini loptica na platformi pomaknut prema slobodnom neučvršćenom vrhu platforme.



Slika 4.19: Pozicija loptice na platformi



Slika 4.20: Pozicija loptice po x i y osi



Slika 4.21: Upravljački signali servomotora

4.8. Rasprava

Ispitane su sve funkcije sustava korištenjem pokaznog algoritma upravljanja. Izrađena glavna aplikacija ispunjava sve svoje funkcionalnosti te se izrađeni sustav može koristiti kao platforma za izradu, proučavanje i ispitivanje različitih algoritama upravljanja.

Kvaliteta upravljanja može se poboljšati na različite načina. Korištena platforma, kao što je istaknuto, zakrivljena je prema slobodnom neučvršćenom vrhu platforme. Razlog zakrivljenosti leži u činjenici da je platforma izrađena od plastičnog materijala koji se s vremenom deformirao. Ploha platforme nije glatka već je uslijed korištenja oštećena te ima sitne udubine i rupice koje stvaraju probleme pri upravljanju. Kvaliteta upravljanja značajno bi se poboljšala ispravljanjem te popravljanim oštećenja koje ploha platforme ima uslijed korištenja. Ovaj problem mogao bi se riješiti postavljanjem tankog staklenog sloja na samu platformu. Ploha na kojoj bi se kretala loptica bila bi glatka i ravna čime bi kvaliteta upravljanja porasla.

Servomotori se upravljaju u otvorenoj petlji tj. sustav zada servomotoru vrijednost u koju želi da se on postavi a on to uradi sam. Korištenjem boljih servomotora za koje bi sustav znao da su dohvatili željenu poziciju kvaliteta upravljanja bi se mogla poboljšati. Korišteni servomotori imaju veliku prednost koju predstavlja njihova cijena i dostupnost no ipak u budućnosti bi valjalo razmisliti o implementaciji aktuatora s naprednijim karakteristikama.

Kvaliteta upravljanja značajno se može poboljšati korištenjem naprednijih algoritama upravljanja kao što su neizraziti (*eng* fuzzy) algoritmi upravljanja, LQR i MPC algoritmi upravljanja, adaptivni algoritmi upravljanja i slično. Izrada takvih algoritama van je dosega ovog rada te nije obrađena.

Rezultati različitih eksperimenata na realnom sustavu dokazuju primjenjivost ovakvog rješenja kao samostalnog i neovisnog edukacijskog modela.

5. Zaključci

Ispitivanje upravljačkih algoritama na stvarnim modelima zahtjevniji je, izazovniji ali i zanimljiviji problem od ispitivanja u simulacijskim okruženjima. Na realnom sustavu projektant algoritama upravljanja susreće se s problemima koje nisu nužno predviđene na simulacijskim modelima (npr. nepredviđeno trenje, oštećenja platforme, različiti uvjeti osvjetljenja koja stvaraju problem algoritmu obrade slike i sl.). Algoritmi upravljanja koji u simulacijskim okruženjima rade savršeno na stvarnom sustavu se ne moraju nužno ponašati dobro, ispitivanje na stvarnom sustavu dakle daje ocjenu projektiranog algoritma upravljanja. Sve navedeno pokazuje da rad na stvarnom sustavu upravljanja kao posljedicu ima puno veće dodane vrijednosti nego rad u simulacijskom okruženju.

S ciljem izrade sustava za ispitivanje naprednih algoritama upravljanja određen je matematički model platforme koji opisuje gibanje platforme i loptice na njoj. Simulacijama u Matlabu ispitana je ispravnost matematičkog modela. Korištena platforma je zakrivljena te je uvedena pomoćna upravljačka petlja po kutu zakrivljenosti platforme. Metodom Truxall-Guillemin projektiran je pokazni regulator korišten pri eksperimentiranju na pravom sustavu.

Sustav je realiziran korištenjem odgovarajućeg sklopovlja i razvijene programske podrške. Programska podrška koristi se za upravljanje sustavom, realizaciju vizualne povratne veze, iscrtavanje odziva te dodavanje novih algoritama upravljanja jednostavnim programiranjem u realiziranoj glavnoj aplikaciji. Dodavanje novih algoritama omogućeno je bez upotrebe dodatnih programskih razvojnih alata što čini razvijenu aplikaciju samostalnim i neovisnim rješenjem.

Ovaj rad, dakle predstavlja zaokruženu cjelinu razvoja edukacijskog modela koji obuhvaća postupak analize i sinteze sustava u simulacijskom okruženju te implementaciju na realnom sustavu. Realizirani edukacijski model predstavlja samostojeće (*eng* stand alone) rješenje koje svojim korisnicima omogućava ispitivanje i istraživanje klasičnih i naprednih modernih algoritama upravljanja.

Budući rad na području sustava loptice na platformi sastojao bi se od poboljšavanja mehaničkih konstrukcija korištene platforme te implementacija naprednih algoritama upravljanja koji bi poboljšali kvalitetu upravljanja lopticom na platformi.

6. Zahvala

Zahvaljujem se svom mentoru prof. dr. sc. Stjepanu Bogdanu što mi je omogućio izradu ovog zanimljivog rada te što je svojim vodstvom, savjetima, i utrošenim vremenom dao veliku potporu pri izradi istog.

7. Literatura

- [1] Jadlovská, A., Jajčičin, Š., Lonščák, R.: Modelling and PID Control Design of Nonlinear Educational Model Ball & Plate, Editors: Fikar, M., Kvasnica, M., In Proceedings of the 17th International Conference on Process Control '09, Štrbské Pleso, Slovakia, 475–483, 2009.
- [2] Yuan, D.; Zhang, Z.; , "Modelling and control scheme of the ball-plate trajectory-tracking pneumatic system with a touch screen and a rotary cylinder," *Control Theory & Applications, IET* , vol.4, no.4, pp.573-589, April 2010 doi: 10.1049/iet-cta.2008.0540
- [3] Xiucheng Dong; Zhang Zhang; Jiagui Tao; , "Design of Fuzzy Neural Network Controller Optimized by GA for Ball and Plate System," Fuzzy Systems and Knowledge Discovery, 2009. FSKD '09. Sixth International Conference on , vol.4, no., pp.81-85, 14-16 Aug. 2009 doi: 10.1109/FSKD.2009.710
- [4] Huida Duan; Yantao Tian; Guangbin Wang; , "Trajectory tracking controller design of high-order ball and plate system based on improved auto-disturbance rejection controller," Systems and Control in Aeronautics and Astronautics (ISSCAA), 2010 3rd International Symposium on , vol., no., pp.80-85, 8-10 June 2010 doi: 10.1109/ISSCAA.2010.5633233
- [5] Wang Hongrui; Tian Yantao; Fu Siyan; Sui Zhen; , "Nonlinear control for output regulation of ball and plate system," Control Conference, 2008. CCC 2008. 27th Chinese , vol., no., pp.382-387, 16-18 July 2008 doi: 10.1109/CHICC.2008.4605473
- [6] Ming Bai; Huiqiu Lu; Jintao Su; Yantao Tian; , "Motion Control of Ball and Plate System Using Supervisory Fuzzy Controller," Intelligent Control and Automation, 2006. WCICA 2006. The Sixth World Congress on , vol.2, no., pp.8127-8131, 0-0 0 doi: 10.1109/WCICA.2006.1713557
- [7] Knuplez, A.; Chowdhury, A.; Svecko, R.; , "Modeling and control design for the ball and plate system," Industrial Technology, 2003 IEEE International Conference on , vol.2, no., pp. 1064-1067 Vol.2, 10-12 Dec. 2003 doi: 10.1109/ICIT.2003.1290810
- [8] Hongrui Wang; Yantao Tian; Ce Ding; Qing Gu; Feng Guo; , "Output regulation of the ball and plate system with a nonlinear velocity observer," Intelligent Control and Automation, 2008. WCICA 2008. 7th World Congress on , vol., no., pp.2164-2169, 25-27 June 2008 doi: 10.1109/WCICA.2008.4593259
- [9] Moarref, M.; Saadat, M.; Vossoughi, G.; , "Mechatronic design and position control of a novel ball and plate system," Control and Automation, 2008 16th Mediterranean Conference on , vol., no., pp.1071-1076, 25-27 June 2008 doi: 10.1109/MED.2008.4602212
- [10] Buntić, B. Vizualna povratna veza u sustavu upravljanja platformom s dva stupnja slobode. Diplomski rad. FER. 2007.

- [11] Rosić, N. Sustav upravljanja platformom s dva stupnja slobode. Diplomski rad. FER. 2008.
- [12] Pankretić, I. Mikroprocesorski sustav upravljanja platformom s dva stupnja slobode. Diplomski rad. FER. 2008.
- [13] Hoško, I. Obrada slike na mikroprocesorskom sustavu Gumstix. Diplomski rad. FER. 2009.
- [14] Petrović, Ivan. Predavanja iz predmeta „Računalno upravljanje sustavima“.FER. Zagreb 2011.
- [15] Bogdan, Stjepan. Predavanja iz predmeta „Osnove inteligentnog upravljanja“. FER. Zagreb 2011.

- [16] Lynxmotion SSC-32 Servo Controller. <http://www.lynxmotion.com/p-395-ssc-32-servo-controller.aspx>. travanj 2011.
- [17] Webcam C200. <http://www.logitech.com/en-us/webcam-communications/webcams/devices/5865>; travanj 2011.
- [18] Interfacing with a Joystick using C# - CodeProject. <http://www.codeproject.com/KB/directx/joystick.aspx>. Travanj 2011.

8. Sažetak

Autor: Grgo Čupić

Naslov rada: „Loptica na platformi: sustav za ispitivanje naprednih algoritama upravljanja“

Cilj ovog rada je izrada edukacijskog modela za istraživanje i ispitivanje naprednih algoritama upravljanja te ocjenjivanje kvalitete takvog upravljanja. Osnovni zahtjev edukacijskog modela je realizacija samostojećeg (*eng.* stand alone) rješenja. Loptica na platformi u žarištu je istraživanja različitih grupa od kraja 1980ih godina. Loptica koja se pozicionira na platformi primjer je inherentno nestabilnog, nelinearnog te u mnogim slučajevima spregnutog sustava. Navedene karakteristike čine ovaj sustav odličnom podlogom za realizaciju edukacijskog modela.

Razvoj edukacijskog modela se sastoji od analize, sinteze te implementacije na realnom sustavu. Izvedeni su matematički modeli platforme te loptice koja se giba po njoj. Matematički modeli daju podlogu za sintezu algoritma upravljanja te ispitivanje sustava u simulacijskom okruženju Matlab Simulink. Korištena platforma nije u potpunosti ravna i kruta ploha već je izvitoperena u smjeru slobodnog vrha platforme. Zakrivljenost platforme predstavlja dodatan problem pri upravljanju koji se rješava uvođenjem pomoćne upravljačke petlje za linearizaciju platforme. Pokazni algoritam upravljanja projektira se korištenjem metode Truxal-Guillemine. Postupci analize i sinteze sustava popraćeni su odgovarajućim simulacijskim odzivima.

Opis cjelokupnog sustava u simulacijskom okruženju predstavlja temelj za implementaciju na realnom sustavu. Realizirani sustav sastoji se od: platforme pokretane servomotorima s lopticom, potrebnog sklopovlja i programske podrške koju izvodi osobno računalo. Sklopovska podrška uključuje web-kameru, kontroler servomotora i igraću palicu. Programska podrška realizirana je kao Windows aplikacija sa sljedećim funkcijama: korisničko sučelje, realizacija povratne veze sustava preko algoritma obrade slike s web-kamere, realizacija upravljanja sustavom (ručno preko igraće palice ili preko algoritma upravljanja) i iscrtavanje odziva varijabli sustava (pozicija loptice i upravljačkih signala). Najvažnija funkcija razvijene programske podrške je mogućnost dodavanja novih algoritama upravljanja u samoj aplikaciji bez potrebe korištenja alata za razvoj programske podrške. Korisnički upravljački algoritam dinamički se prevodi u stvarnom vremenu (*eng.* Dynamic Compilation).

Ispitane su sve funkcije sustava korištenjem pokaznog algoritma. Rezultati različitih eksperimenata na realnom sustavu dokazuju primjenjivost ovakvog rješenja kao samostalnog i neovisnog edukacijskog modela.

Ključne riječi: nelinearni sustav upravljanja, sustav loptice na platformi, vizualna povratna veza, dinamičko prevođenje, edukacijski model

9. Summary

Author: Grgo Čupić

Research paper title: „Ball and Plate: Advanced Controller Design Testing System”

The main objective of the research described in this paper is to develop an educational model designed for studying, experimenting and evaluating different advanced controller designs based on classical and modern control theory. The developed educational model has to meet the basic requirement of being a standalone solution. Since the late 1980s the ball and plate system has been utilized by numerous research groups. The ball and plate is a classic example of inherently unstable, non-linear and in many cases coupled system. Considering the given characteristics the ball and plate system makes an excellent basis for an educational model.

The development of educational model consists of three basic steps: system modeling, controller design and real system implementation. The paper describes derived mathematical models representing the ball and plate system. Mathematical models provide a basis for controller designing process and system testing using Matlab Simulink environment. The plate utilized is not completely flat and solid; its surface is warped via the direction of the unconstrained plate point. The plate surface curvature is an additional control problem, since it introduces more non-linearities in the system. The plate surface curvature problem is solved by introducing auxiliary closed control loop based on the inverted surface mathematical model. The Truxal-Guillemin method is used for the demo controller design. The proceedings of system modeling and controller design are accompanied by corresponding simulation responses.

Simulation environment system representation presents the basis of real system implementation. The system realized in this paper comprises of: two degree of freedom servomotor driven platform with ball, supporting hardware and software performed on a Windows based personal computer. The supporting hardware consists of: web-cam, servomotor controller and a joystick. The developed software application is Windows based and it features: user interface, web-cam feedback using image processing algorithm, real system control (manually via joystick or in automatic mode using designed controller) and system response plotting (ball position and system control signal). The most significant software feature is the new controller design adding ability without the need of other software development tools. The user controller design is dynamically compiled in real time.

All of the system features have been tested using the demo controller design. Various experiment results prove the applicability of the proposed advanced controller design testing system as an autonomous and independent educational model.

Key words: non-linear control system, ball and plate system, web-cam feedback, dynamic compilation, educational model

10. Kratki životopis

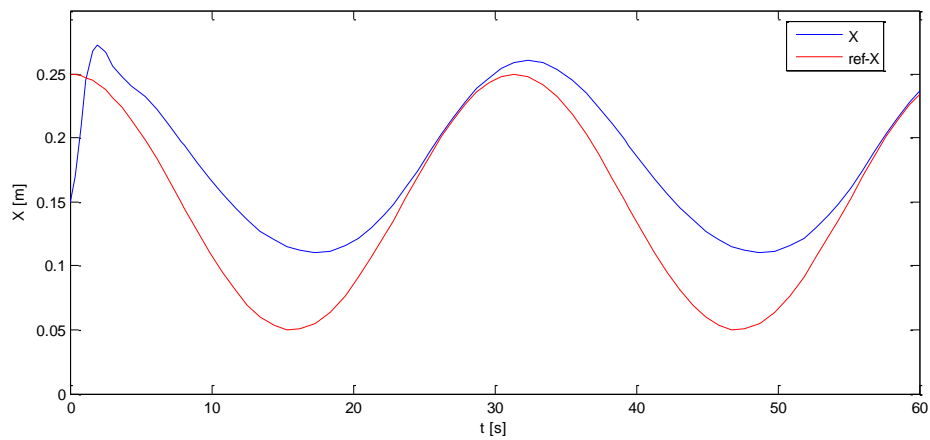
Grgo Čupić rođen je 15.10.1987. u Splitu. Godine 2002. upisao je III. prirodoslovno-matematičku gimnaziju u Splitu. Maturirao je s odličnim uspjehom 2006. godine. Iste godine upisuje preddiplomski studij elektrotehnike i informacijske tehnologije pri Fakultetu elektrotehnike i računarstva.

Završetkom preddiplomskog studija na smjeru Automatika 2009. dodjeljuje mu se naziv sveučilišnog prvostupnika (baccalaureus) inženjera elektrotehnike i informacijske tehnologije. Nastavlja studij na diplomskom studiju Automatike te se trenutno radi na izradi diplomskog rada.

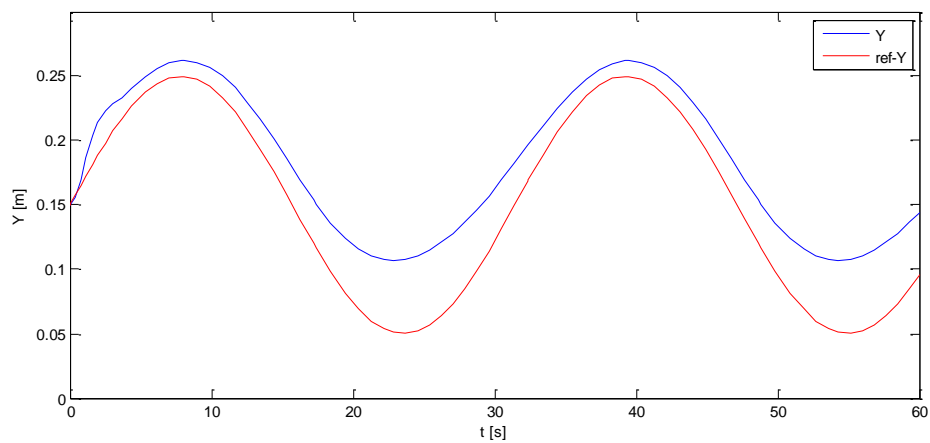
Godine 2010. osvaja 2. mjesto na natječaju „Werner von Siemens“ za najbolje studentske radove u akademskoj godini 2009./2010. na FER-u s primjenom u područjima industrije, energije i zdravstvene skrbi tvrtke Siemens d.o.o., s radom „Nosivi bežični sustav za mjerenje pokretljivosti i detekciju pada starijih osoba“.

U akademskoj godini 2010./2011. paralelno studiju na FER-u vanredno upisuje diplomski studij Poslovne ekonomije pri Ekonomskom fakultetu u Zagrebu.

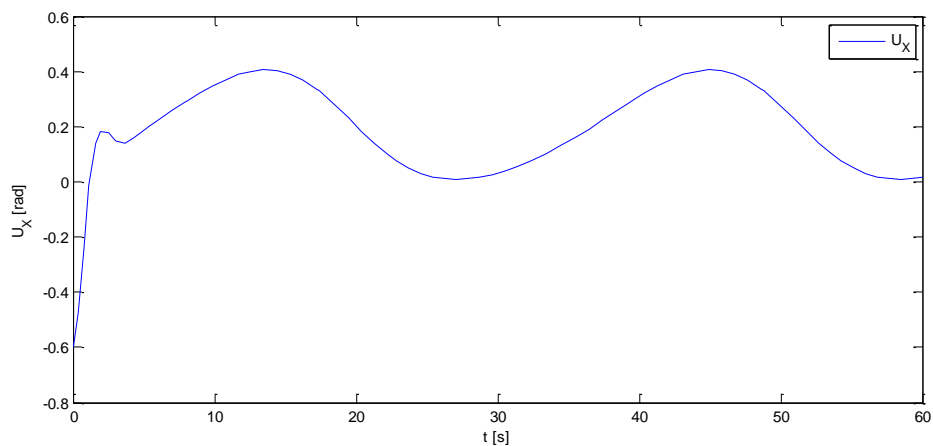
Prilog A Simulacijski odzivi - P regulator bez pomoćne upravljačke petlje



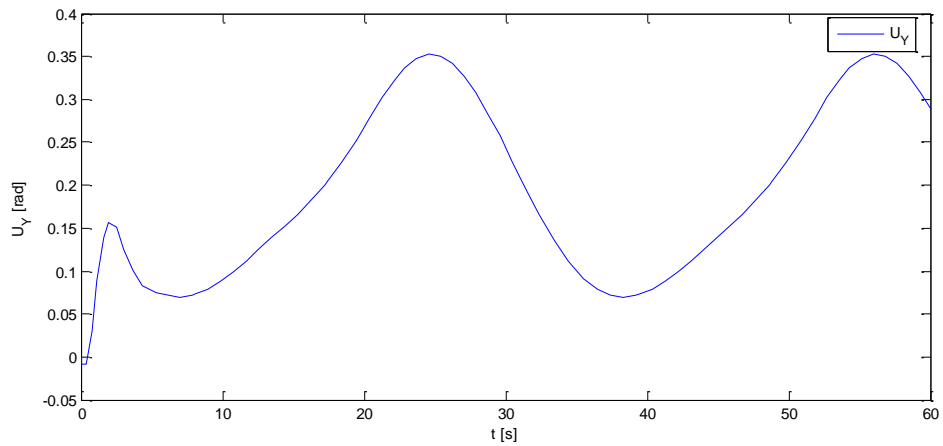
Slika A.1: Pozicija loptice po x osi uz korištenje P regulatora bez linearizacije neravnina



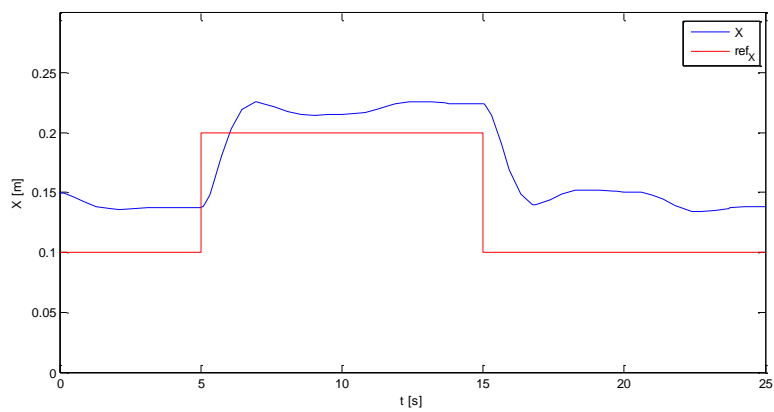
Slika A.2: Pozicija loptice po y osi uz korištenje P regulatora bez linearizacije neravnina



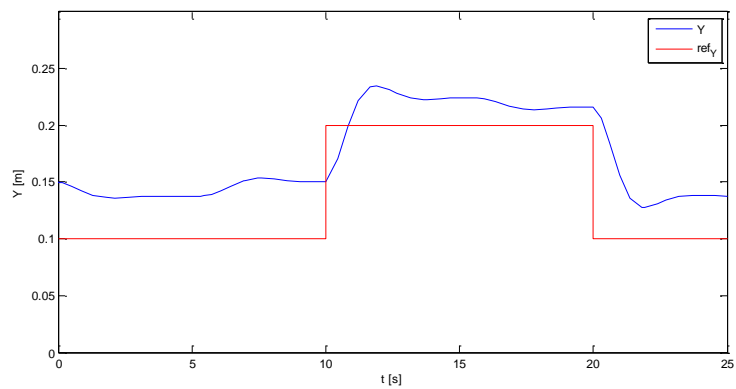
Slika A.3: Upravljački signal na izlazu iz P regulatora za x os bez linearizacije neravnina



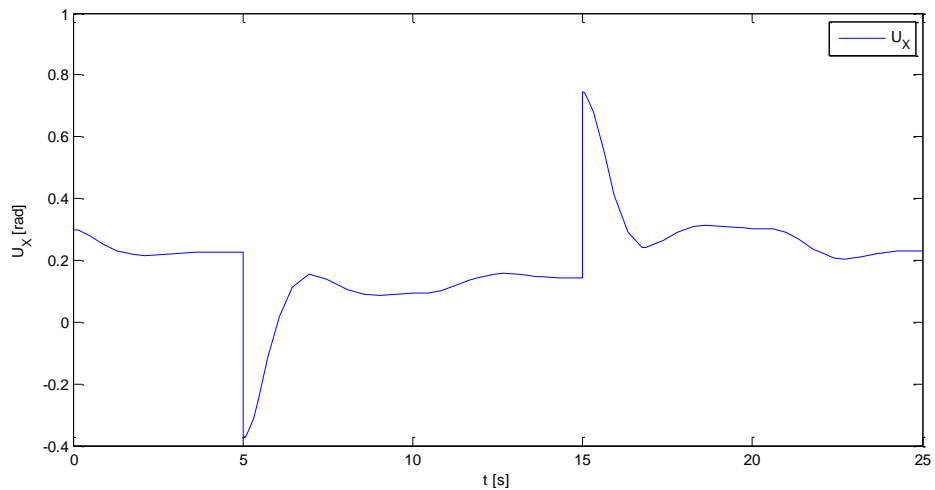
Slika A.4: Upravljački signal na izlazu iz P regulatora za y os bez linearizacije neravnina



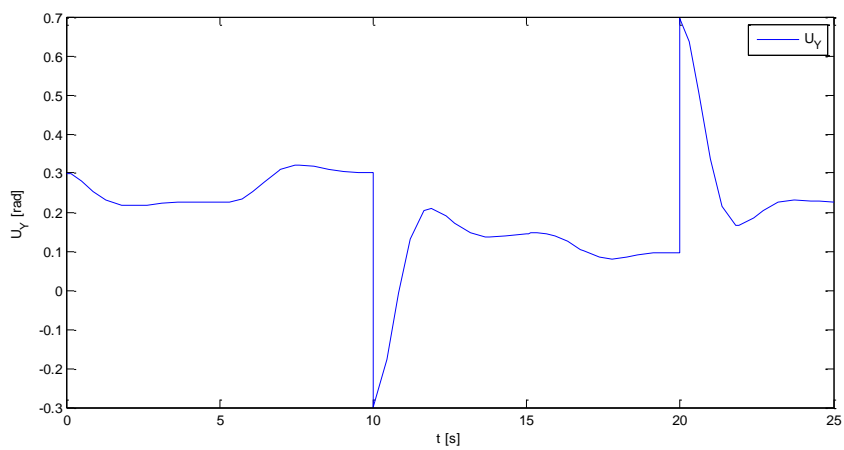
Slika A.5: Pozicija loptice po x osi uz korištenje P regulatora bez linearizacije neravnina



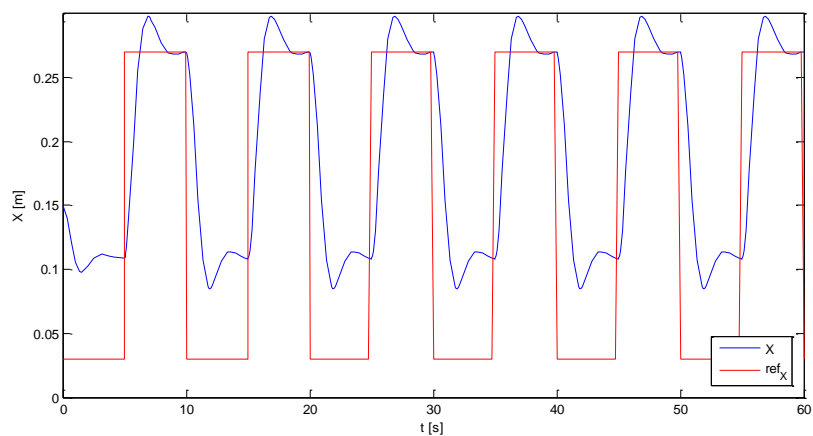
Slika A.6: Pozicija loptice po y osi uz korištenje P regulatora bez linearizacije neravnina



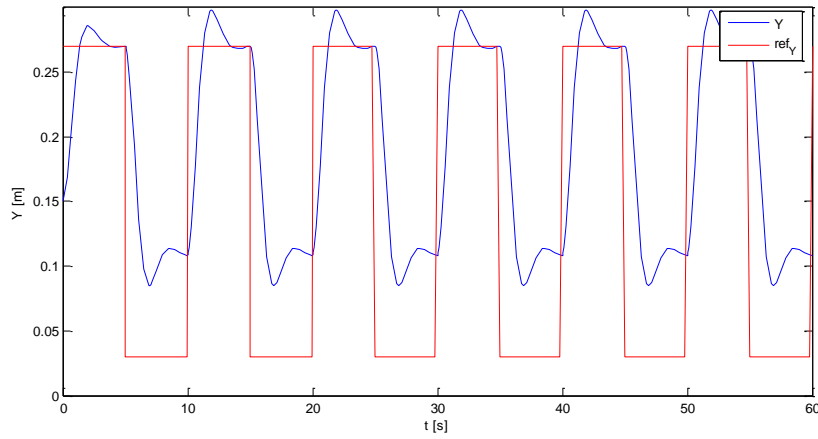
Slika A.7: Upravljački signal na izlazu iz P regulatora za x os bez linearizacije neravnina



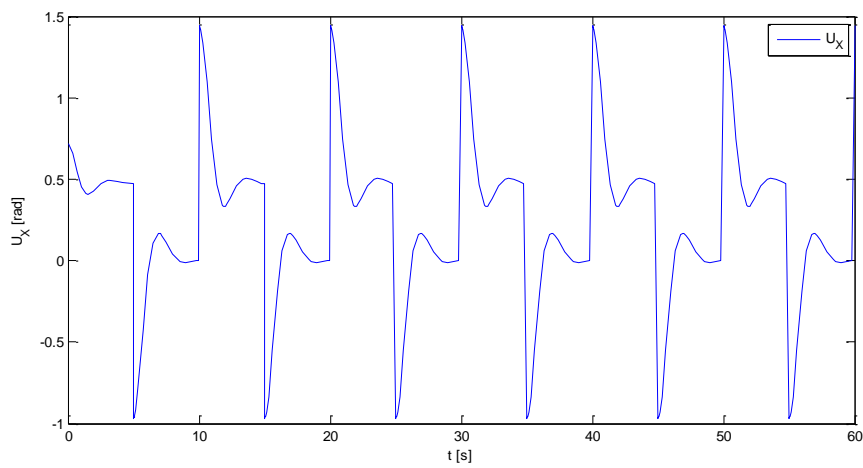
Slika A.8: Upravljački signal na izlazu iz P regulatora za y os bez linearizacije neravnina



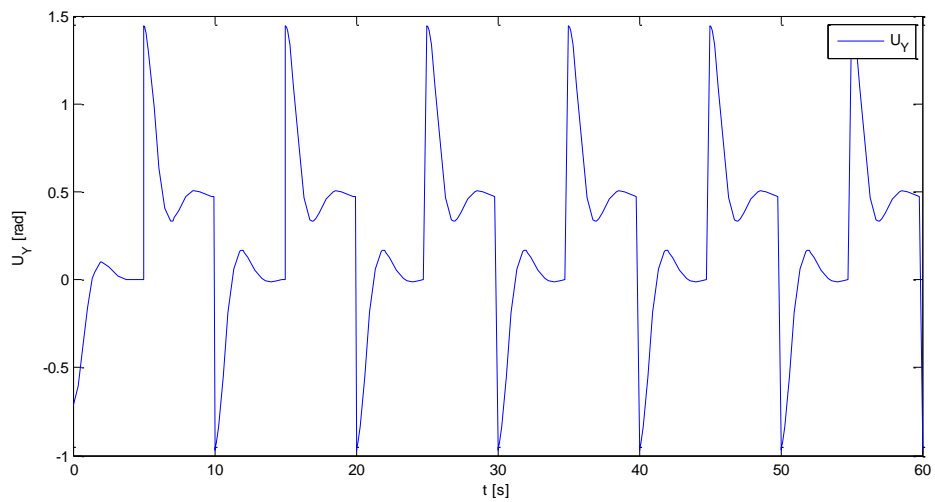
Slika A.9: Pozicija loptice po x osi uz korištenje P regulatora bez linearizacije neravnina



Slika A.10: Pozicija loptice po y osi uz korištenje P regulatora bez linearizacije neravnina

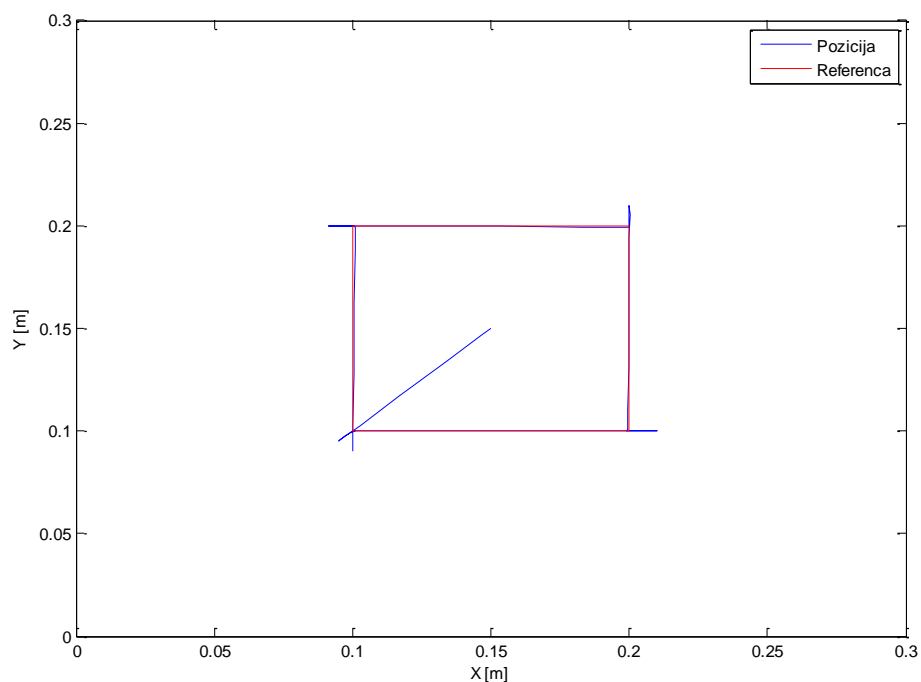


Slika A.11: Upravljački signal na izlazu iz P regulatora za x os bez linearizacije neravnina

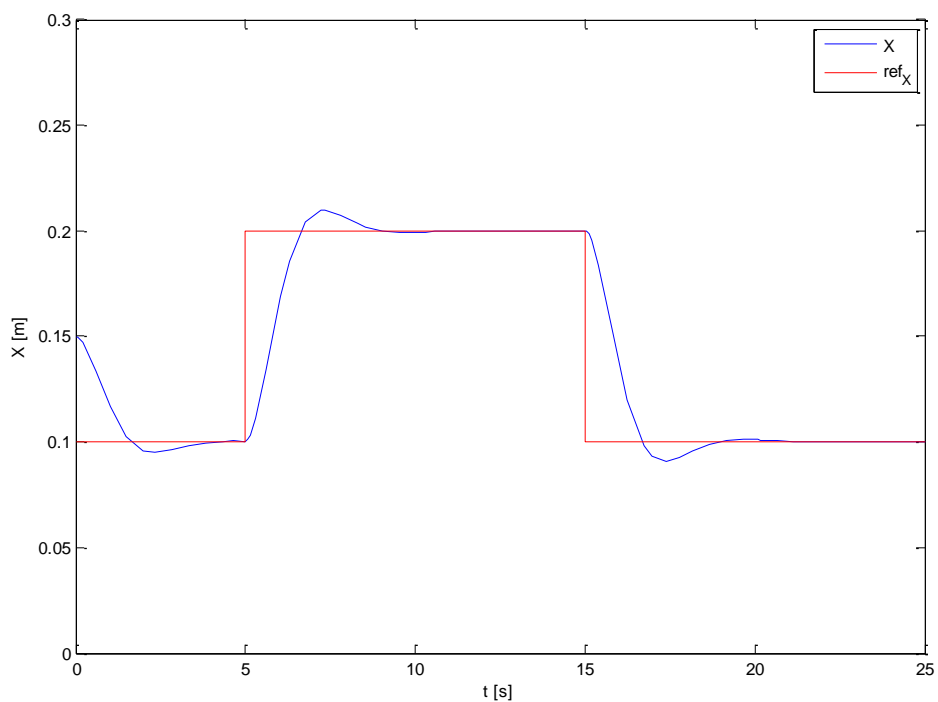


Slika A.12: Upravljački signal na izlazu iz P regulatora za x os bez linearizacije neravnina

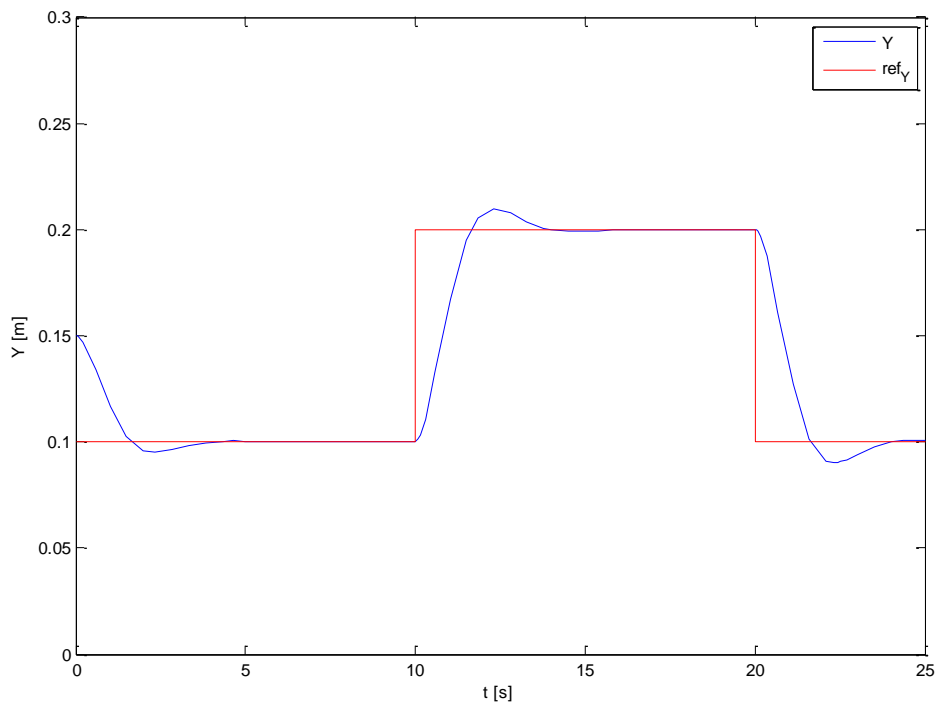
Prilog B Simulacijski odzivi - P regulator sa pomoćnom upravljačkom petljom



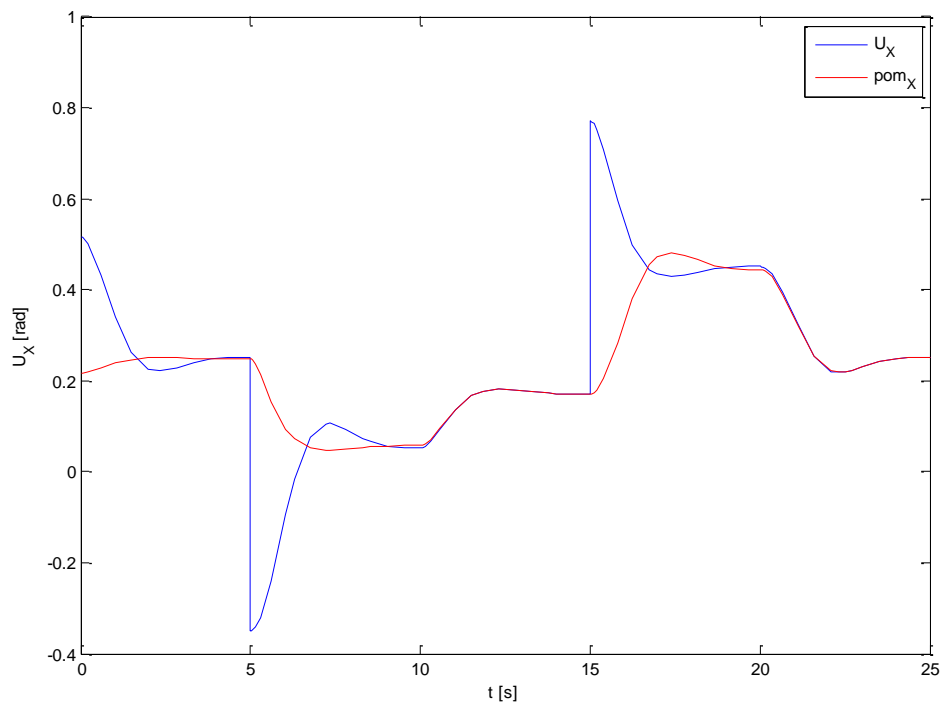
Slika B.1: Pozicija loptice na platformi za simulaciju kvadrata



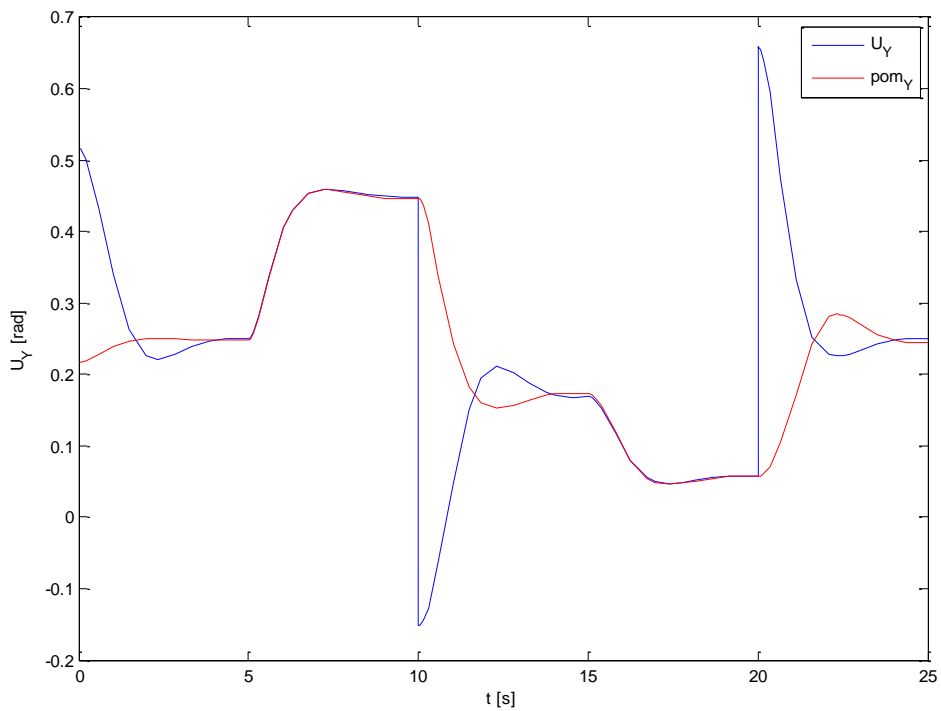
Slika B.2: Pozicija loptice po x osi uz korištenje P regulatora sa linearizacijom neravnina



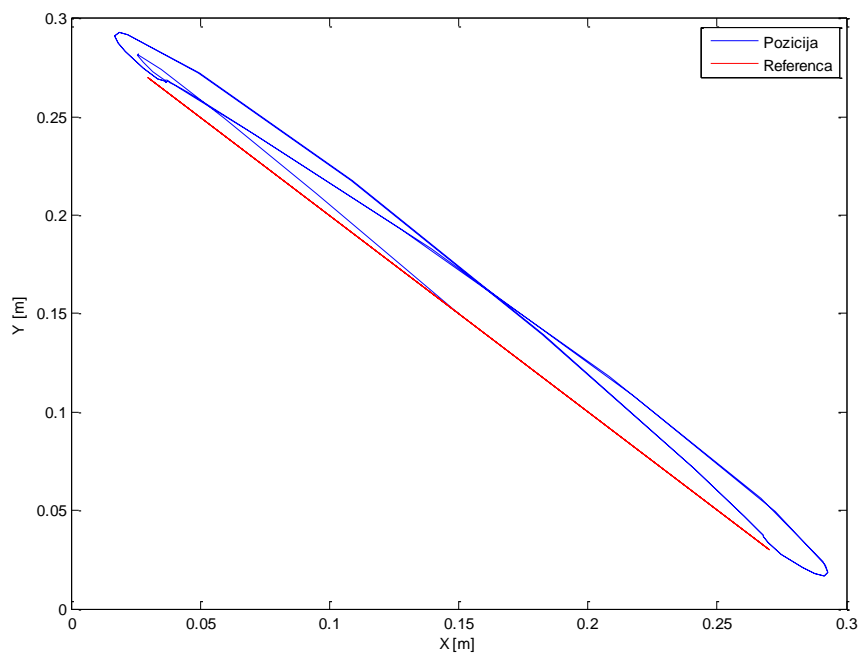
Slika B.3: Pozicija loptice po y osi uz korištenje P regulatora sa linearizacijom neravnina



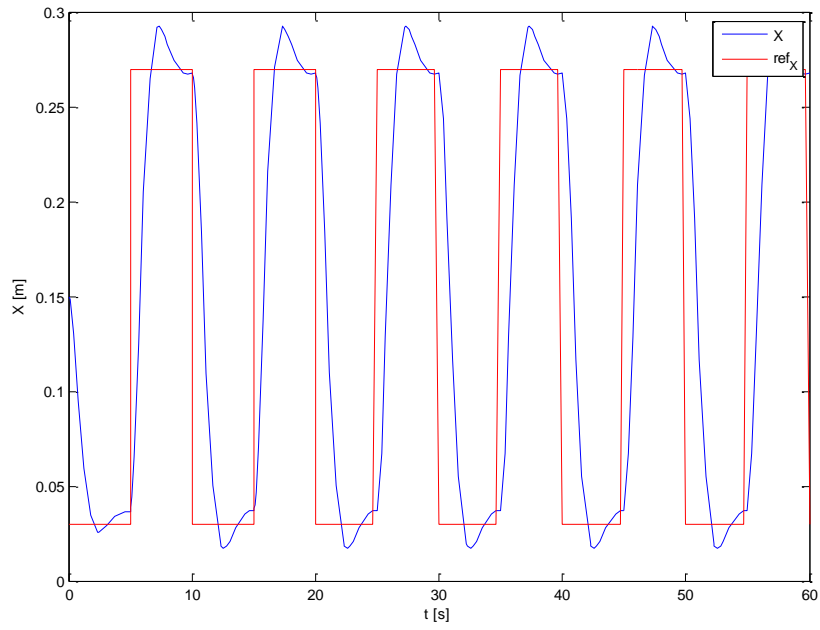
Slika B.4: Upravljački signal na ulazu u sustav za x os uz korištenje P regulatora sa linearizacijom neravnina



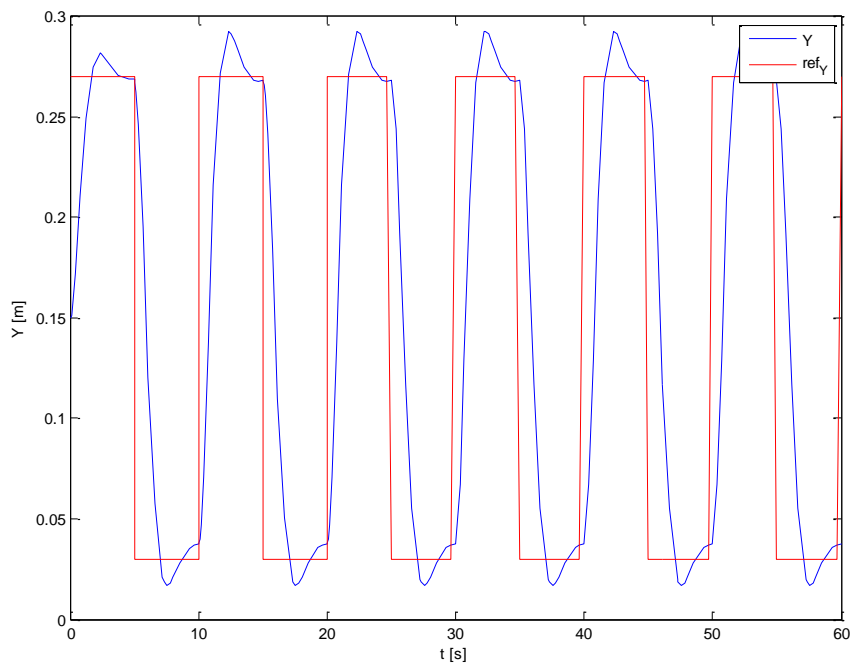
Slika B.5: Upravljački signal na ulazu u sustav za y os uz korištenje P regulatora sa linearizacijom neravnina



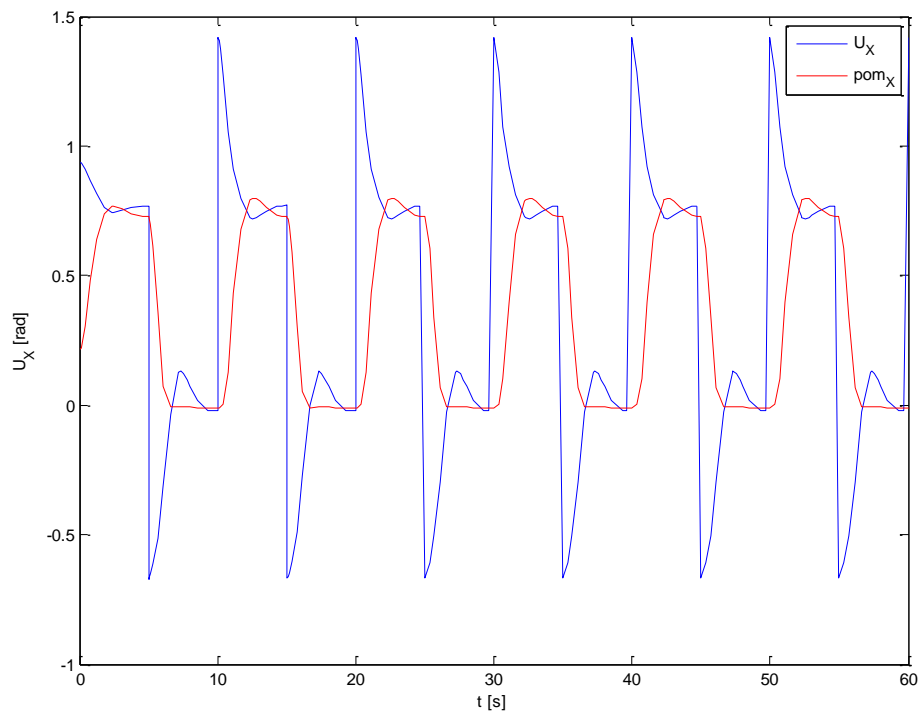
Slika B.6: Pozicija loptice na platformi za simulaciju pravca



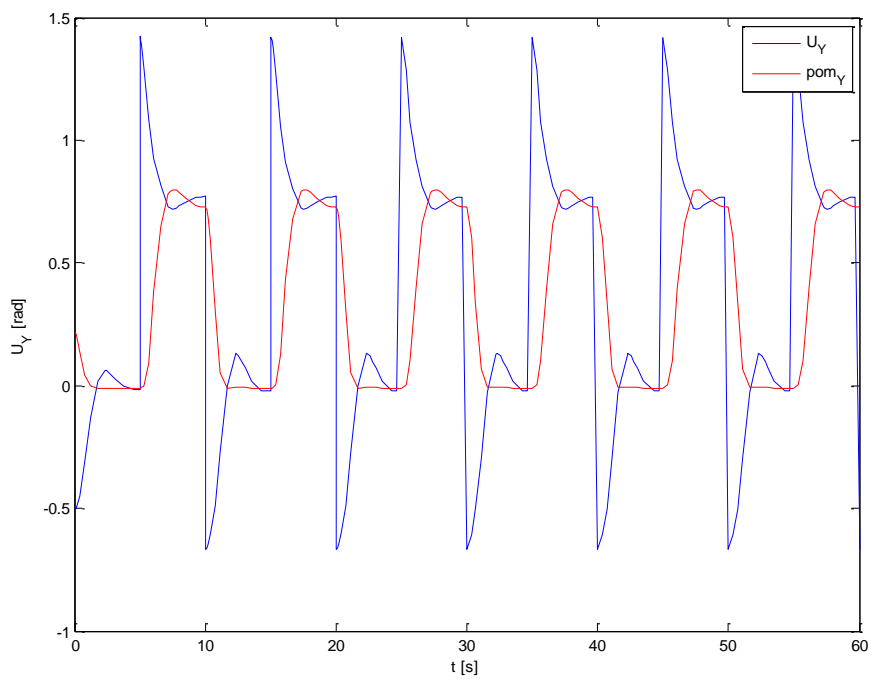
Slika B.7: Pozicija loptice po x osi uz korištenje P regulatora sa linearizacijom neravnina



Slika B.8: Pozicija loptice po y osi uz korištenje P regulatora sa linearizacijom neravnina

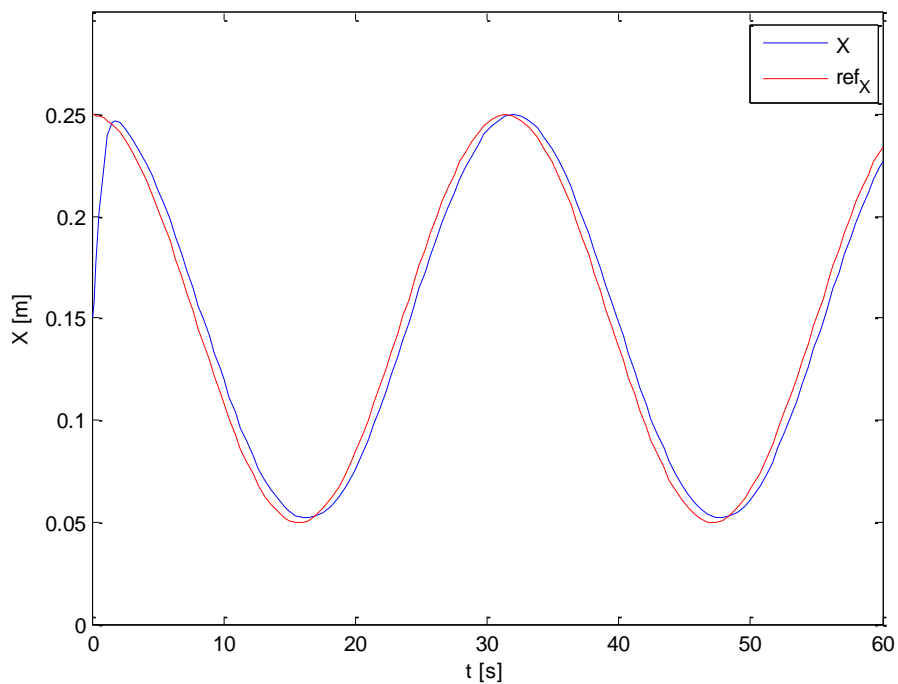


Slika B.9: Upravljački signal na ulazu u sustav za x os uz korištenje P regulatora sa linearizacijom neravnina

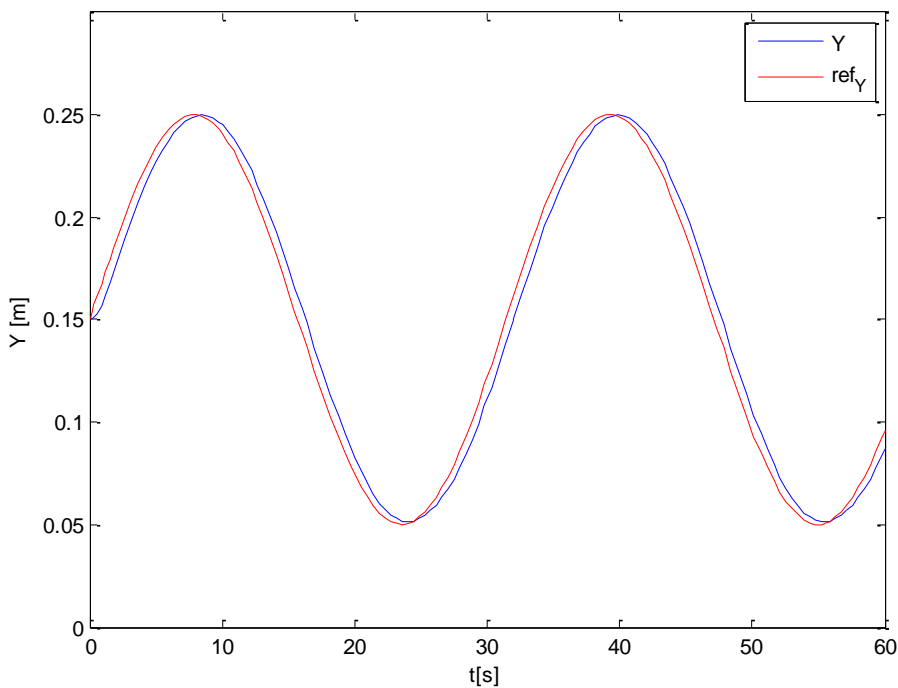


Slika B.10: Upravljački signal na ulazu u sustav za y os uz korištenje P regulatora sa linearizacijom neravnina

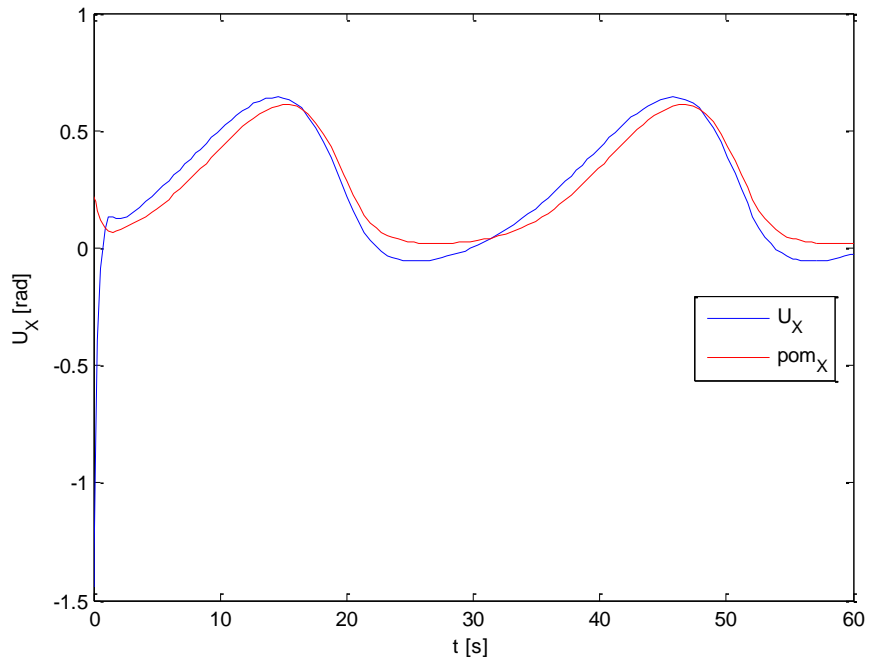
Prilog C Simulacijski odzivi - projektirani regulator sa pomoćnom upravljačkom petljom



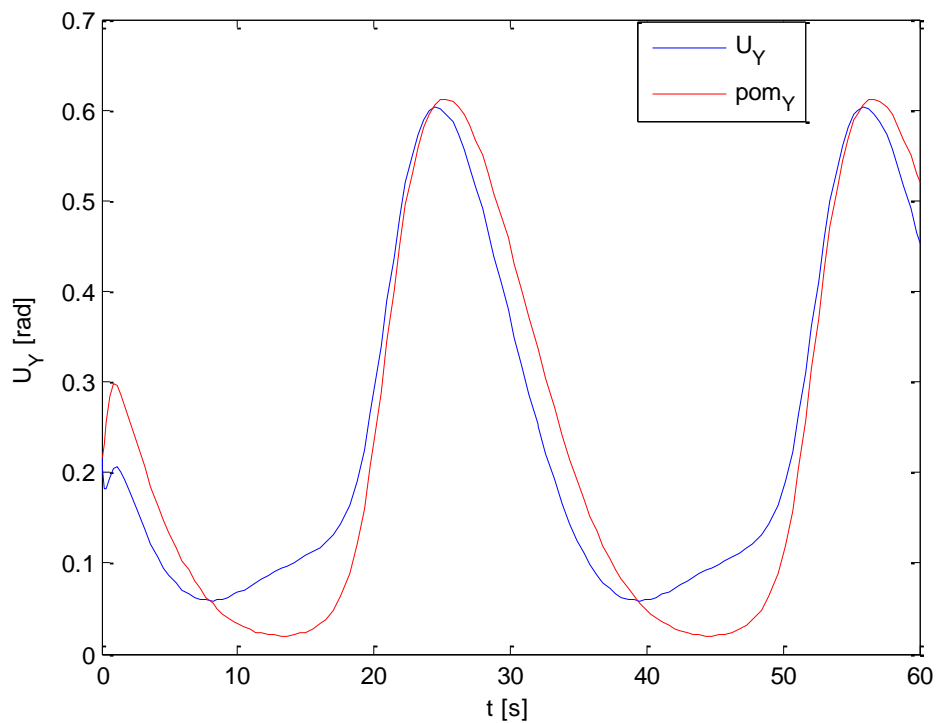
Slika C.1: Pozicija loptice po x osi uz korištenje projektiranog regulatora sa linearizacijom neravnina



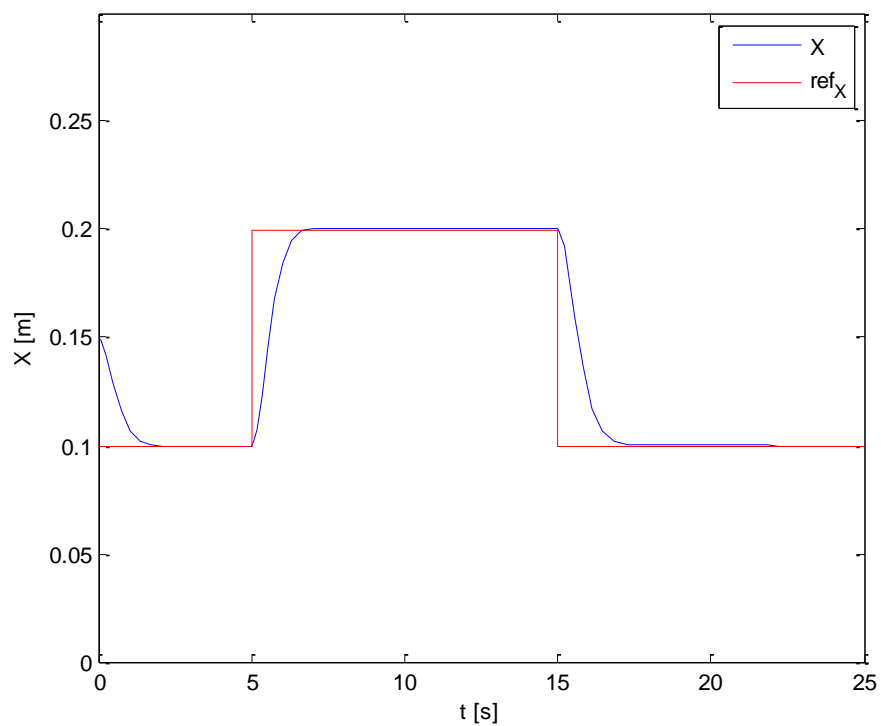
Slika C.2: Pozicija loptice po y osi uz korištenje projektiranog regulatora sa linearizacijom neravnina



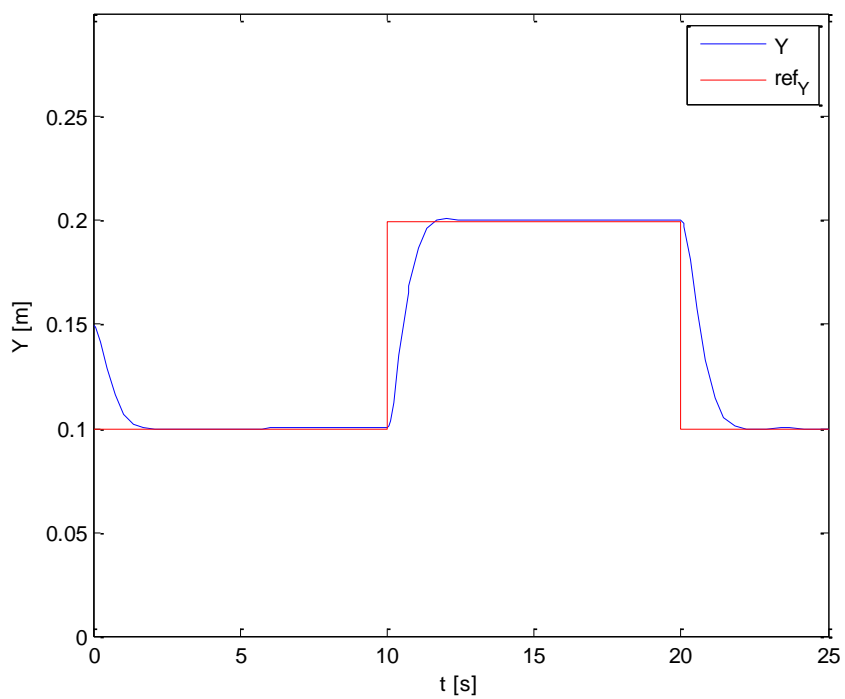
Slika C.3: Upravljački signal na ulazu u sustav za x os uz korištenje projektiranog regulatora sa linearizacijom neravnina



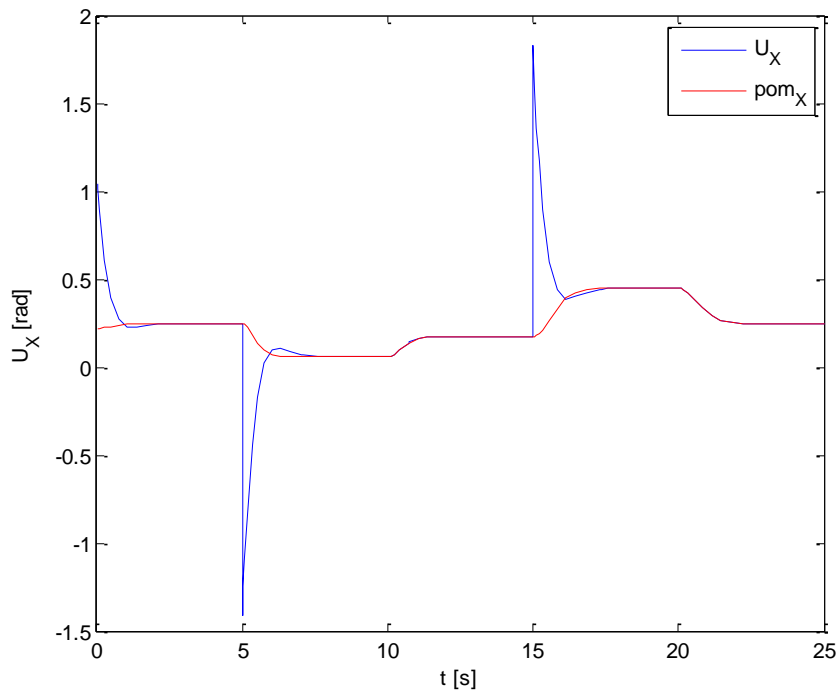
Slika C.4: Upravljački signal na ulazu u sustav za y os uz korištenje projektiranog regulatora sa linearizacijom neravnina



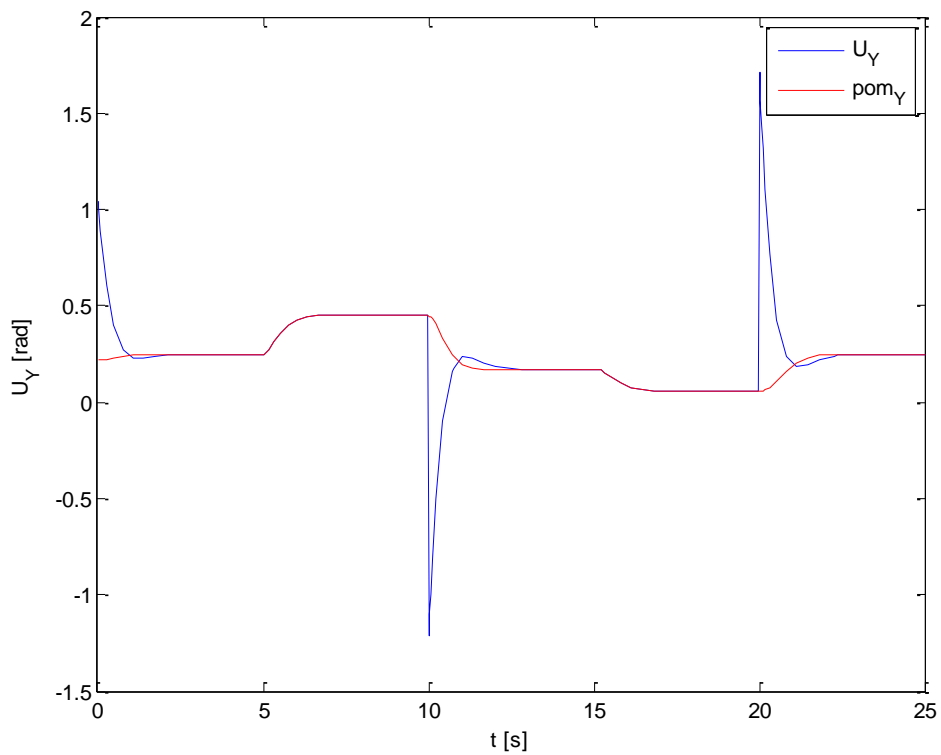
Slika C.5: Pozicija loptice po x osi uz korištenje projektiranog regulatora sa linearizacijom neravnina



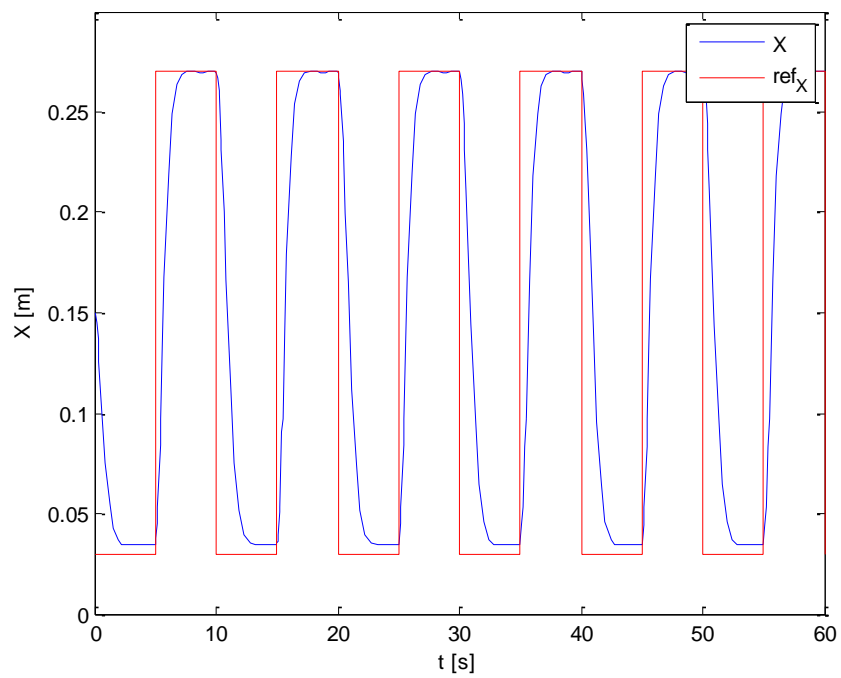
Slika C.6: Pozicija loptice po y osi uz korištenje projektiranog regulatora sa linearizacijom neravnina



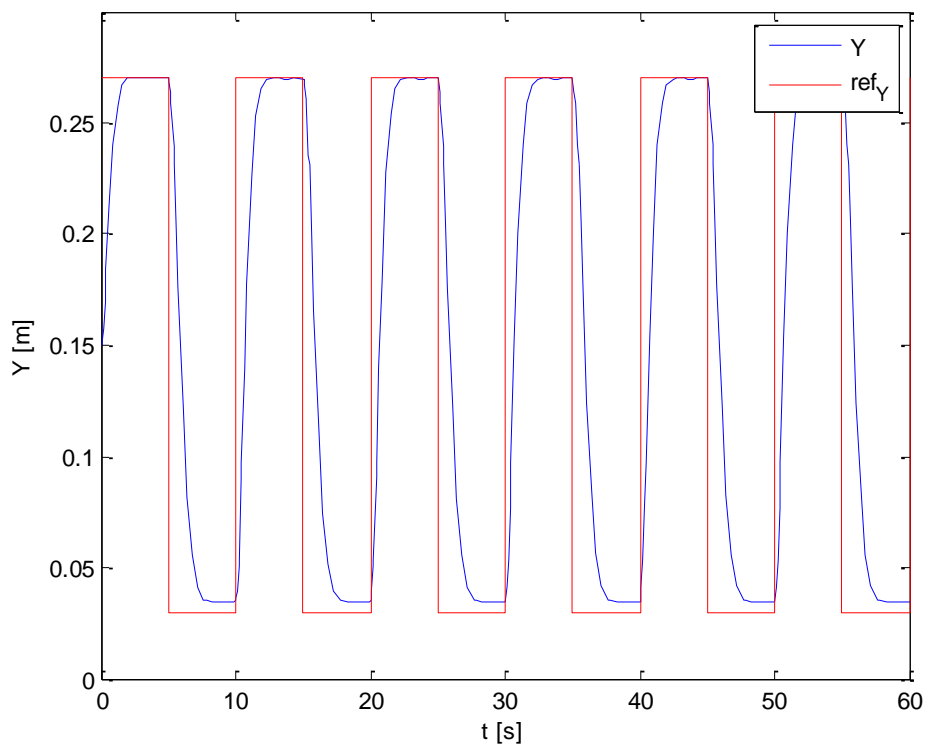
Slika C.7: Upravljački signal na ulazu u sustav za x os uz korištenje projektiranog regulatora sa linearizacijom neravnina



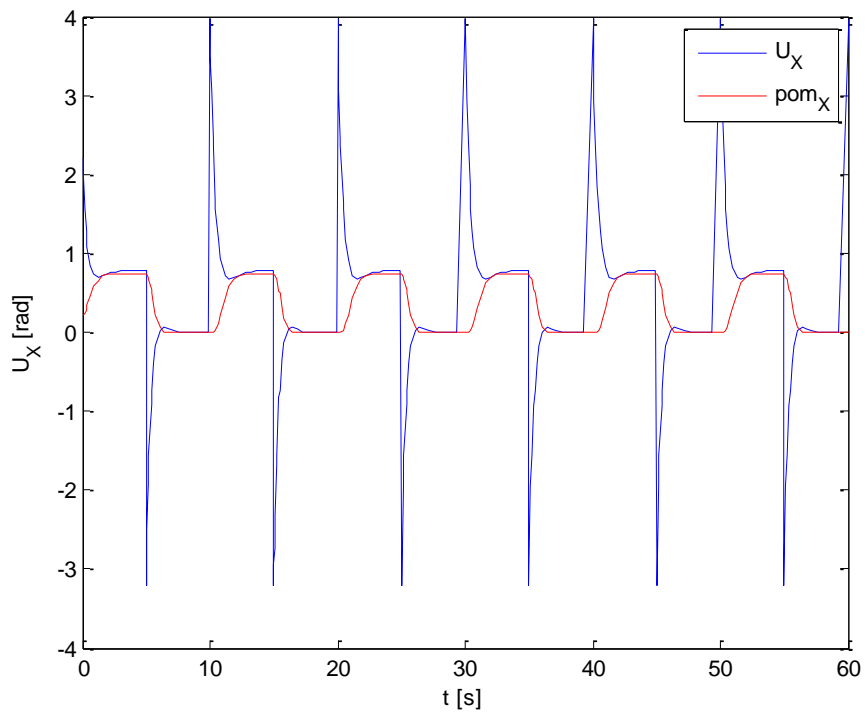
Slika C.8: Upravljački signal na ulazu u sustav za y os uz korištenje projektiranog regulatora sa linearizacijom neravnina



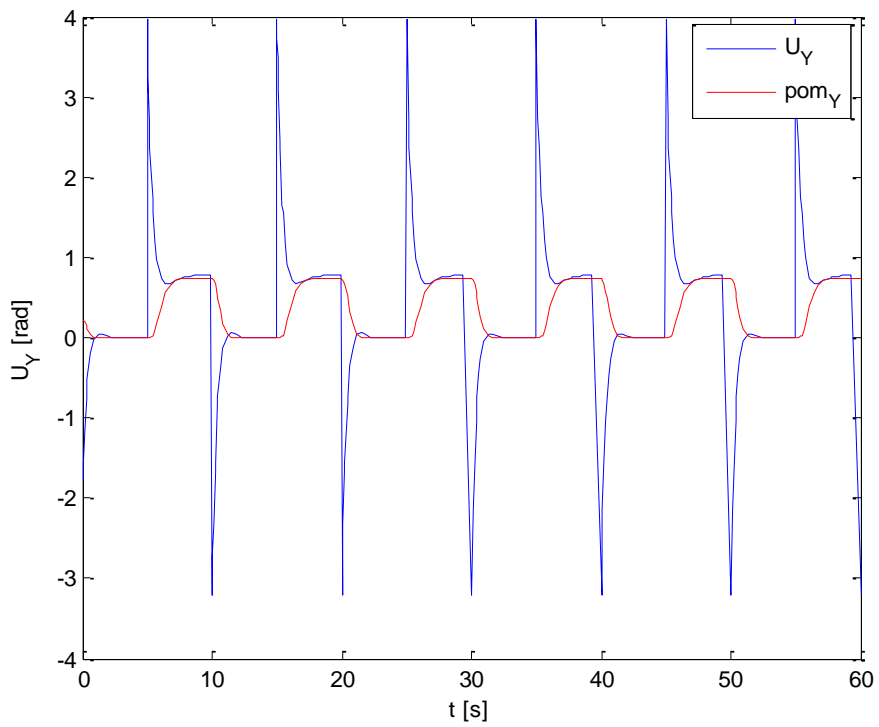
Slika C.9: Pozicija loptice po x osi uz korištenje projektiranog regulatora sa linearizacijom neravnina



Slika C.10: Pozicija loptice po y osi uz korištenje projektiranog regulatora sa linearizacijom neravnina



Slika C.11: Upravljački signal na ulazu u sustav za x os uz korištenje projektiranog regulatora sa linearizacijom neravnina



Slika C.12: Upravljački signal na ulazu u sustav za y os uz korištenje projektiranog regulatora sa linearizacijom neravnina

Prilog D Eksperimentalno korišteni pokazni algoritam upravljanja

```
using System;

public class Regulator
{
    static double x_error_prosli=0;
    static double y_error_prosli=0;

    public double [] algoritam(double x_error, double y_error, double
x_mjereni, double y_mjereni)
    {
        double[] regulator=new double[2];
        double reg_temp_x, reg_temp_y;
        double temp, temp2;
        double x_kut, y_kut;

        reg_temp_x = -1 * 50.0 * x_error + 40.0 * x_error_prosli;
        reg_temp_y = -1 * 50.0 * y_error + 40.0 * y_error_prosli;

        temp = (x_mjereni / 1000.0) * (y_mjereni / 1000.0) * 0.0574073
* 2.0 + 0.0574073 * (y_mjereni / 1000.0) * (y_mjereni / 1000.0) -
0.0631629 * (1 - (x_mjereni / 1000.0) / Math.Sqrt((y_mjereni /
1000.0) * (y_mjereni / 1000.0) + (x_mjereni / 1000.0) * (x_mjereni /
1000.0)));

        temp2 = (x_mjereni / 1000.0) * (y_mjereni / 1000.0) * 0.0574073
* 2.0 + 0.0574073 * (x_mjereni / 1000.0) * (x_mjereni / 1000.0) -
0.0631629 * (1 - (y_mjereni / 1000.0) / Math.Sqrt((y_mjereni /
1000.0) * (y_mjereni / 1000.0) + (x_mjereni / 1000.0) * (x_mjereni /
1000.0)));

        x_kut = Math.Atan(temp);
        y_kut = Math.Atan(temp2);

        regulator[0] = reg_temp_x - 15.0 * x_kut;
        regulator[1] = reg_temp_y - 15.0 * y_kut;

        x_error_prosli=x_error;
        y_error_prosli=y_error;
        return regulator;
    }
}
```