Primjena programskog sustava Matlab/Simulink u projektiranju mehatroničkih sustava

Jamić, Nikola

Undergraduate thesis / Završni rad

2024

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Electrical Engineering and Computing / Sveučilište u Zagrebu, Fakultet elektrotehnike i računarstva**

Permanent link / Trajna poveznica: https://urn.nsk.hr/urn:nbn:hr:168:601526

Rights / Prava: In copyright / Zaštićeno autorskim pravom.

Download date / Datum preuzimanja: **2025-03-22**

Repository / Repozitorij:

FER Repository - University of Zagreb Faculty of Electrical Engineering and Computing repozitory

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 1687

PRIMJENA PROGRAMSKOG SUSTAVA MATLAB/SIMULINK U PROJEKTIRANJU MEHATRONIČKIH SUSTAVA

Nikola Jamić

Zagreb, lipanj 2024.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

ZAVRŠNI RAD br. 1687

PRIMJENA PROGRAMSKOG SUSTAVA MATLAB/SIMULINK U PROJEKTIRANJU MEHATRONIČKIH SUSTAVA

Nikola Jamić

Zagreb, lipanj 2024.

SVEUČILIŠTE U ZAGREBU
FAKULTET ELEKTROTEHNIKE I RAČUNARSTVA

Zagreb, 4. ožujka 2024.

ZAVRŠNI ZADATAK br. 1687

Pristupnik:	**Nikola Jamić (0036529431)**
Studij:	Elektrotehnika i informacijska tehnologija i Računarstvo
Modul:	Računarstvo
Mentor:	prof. dr. sc. Jadranko Matuško

Zadatak: **Primjena programskog sustava Matlab/Simulink u projektiranju mehatroničkih sustava**

Opis zadatka:

Mehatronički sustavi zbog svoje inherentne isprepletenosti različitih inženjerskih domena predstavljaju značajan izazov sa stajališta njihovog projektiranja, a napose se to odnosi na podršku cjelokupnom procesu projektiranja, od projektiranja mehaničkog dijela sustava do implementacije samog upravljačkog algoritma. U radu je potrebno analizirati alate koje su raspoloživi unutar programskog sustava Matab/Simulink u svrhu računalno podržanog projektiranja mehatroničkih sustava. Na primjeru laboratorijskog postava obrnutog njihala demonstrirati korištenje navedenih alata s posebnim naglaskom na generiranje upravljačkog koda i implementaciju upravljačkih algoritama na ugradbene računalne platforme.

Rok za predaju rada: 14. lipnja 2024.

Sadržaj

Uvod 1
1. Mehanički sustavi i mehatronika: Pregled i izazovi 4
1.1. Definicija mehaničkih sustava i mehatronike 4
1.2. Izazovi u mehatronici 4
1.2.1. Integracija različitih inženjerskih disciplina 4
1.2.2. Složenost dizajna 5
1.2.3. Potreba za preciznom kontrolom i automatizacijom 5
2. Matlab/Simulink u mehatroničkom inženjerstvu 6
2.1. Osnovne funkcionalnosti Matlab/Simulinka 6
2.1.1. Opis Matlab/Simulinka 6
2.1.2. Specifični alati i značajke za mehatroniku 6
2.2. Primjena Matlab/Simulinka u mehatronici 9
2.2.1. Modeliranje i simulacija mehatroničkih sustava 9
2.2.2. Razvoj upravljačkih algoritama 9
2.2.3. Integracija i testiranje 10
3. Studija slučaja: Laboratorijski postav inverznog njihala 11
3.1. Matematički model rotacijskog inverznog njihala 11
3.1.1. Detaljan opis postava 12
3.2. Potreba Matlab/Simulinka u studiji slučaja 13
3.2.1. Modeliranje i simulacija 13
3.2.2. Razvoj i testiranje upravljačkih algoritama 13
3.2.3. Implementacija i validacija 13
4. Analiza simulacije rotacijskog inverznog njihala 15
4.1. Simulink model rotacijskog inverznog njihala 15
4.2. Izračun optimalnog pojačanja K 15
4.2.1. Linearno-kvadratni regulator (LQR) 16
4.3. Simulacija sustava stabilizacije inverznog njihala 17
4.4. Simulacija sustava s impulsom 20
4.4.1. Simulacija stabilnog vladanja sustava 20
4.4.2. Simulacija nestabilnog vladanja sustava 22
5. Analiza postava rotacijskog inverznog njihala 25
5.1. Simulink model postava rotacijskog inverznog njihala 25
5.2. Povezivanje postava s računalom pomoću Simulink-a 25
5.3. Analiza podataka dobivenih s postava rotacijskog inverznog njihala 26

5.4. Usporedba rezultata simulacije i stvarnog postava 28
6. Zaključak 30
Literatura 31
Sažetak 32
Summary 33
Privitak 34
Upute za korištenje programske podrške 34
Pokretanje simulacije 34

Uvod

Mehatronika je interdisciplinarno područje inženjerstva. Iz samog naziva uvedenog u Japanu 1969.-e zaključujemo da u njemu sudjeluju i isprepliću se strojarske, elektroničke, informatičke, ali i druge discipline. Od 1982. godine pojam mehatronike prihvaćen je u tehničkom svijetu. Tek je kasnije povezan s računalnim znanostima te je u posljednjih nekoliko godina uveden u razne discipline cijeloga svijeta. Predstavlja cjelovit pristup u inženjerskoj proizvodnji, projektiranju, održavanju i vođenju uređaja, strojeva, naprava i alata te prerasta u novu tehničku filozofiju.

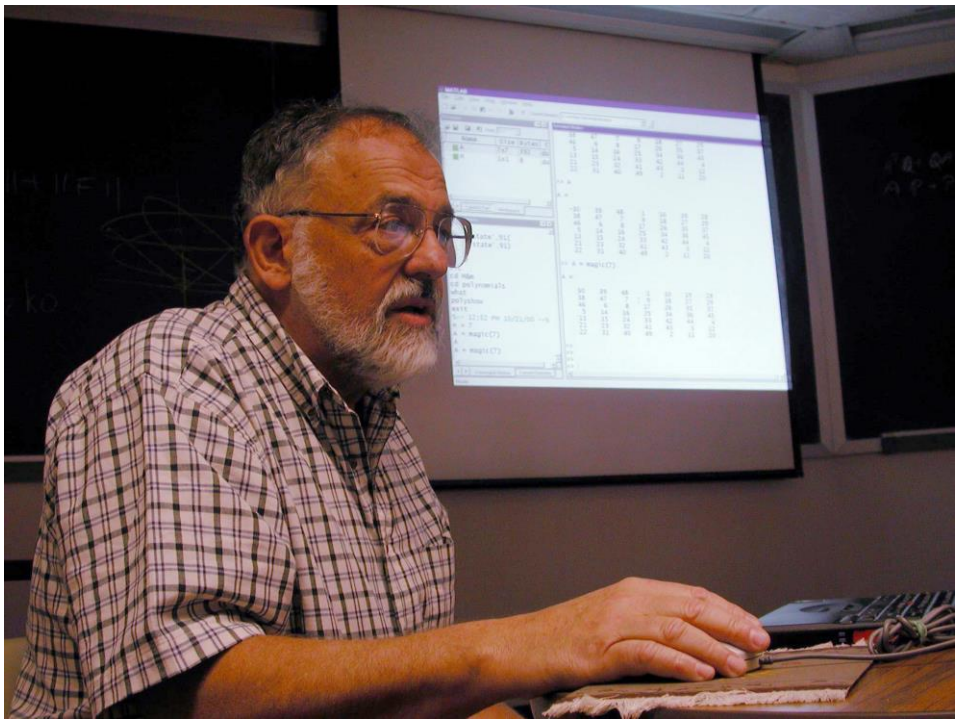
U suvremenom svijetu svakodnevno smo u interakciji s automatiziranim strojevima: od onih jednostavnih poput televizora, mobitela, aparata za kavu, kuhala za vodu do onih najsloženijih poput raznih robota, automobila, i automatskih linija. Svaki taj automatizirani stroj sastoji se od tri osnovne skupine elemenata: senzora, procesora i aktuatora koji zajedno čine jedinstven sustav. Stoga stručnjaci koji se bave mehatronikom moraju imati objedinjena znanja iz područja mehanike, senzorike, elektronike, upravljanja, vođenja i programiranja. U gotovo svim granama inženjerstva, mehatronički sustavi predstavljaju kritičnu sastavnicu napretka u brojnim industrijskim i istraživačkim poljima i zato možemo reći da mehatronika predstavlja temelj za inovacije u brojnim područjima poput automatizacije, robotike, medicine, itd.

Slika 0.1 Područja koje obuhvaća mehatronika

Ovaj rad fokusira se na razmatranje ključne uloge koju Matlab/Simulink igra u procesu projektiranja mehatroničkih sustava. Pomaže inženjerima anticipirati potencijalne probleme, optimizirati dizajn i performanse, te ubrzati razvojni ciklus proizvoda. Analizirajući ovu platformu i njene alate, rad će istražiti kako se Matlab/Simulink koristi za podršku cjelokupnom procesu projektiranja - od kreiranja mehaničke strukture sustava do implementacije sofisticiranih upravljačkih algoritama.

Matlab, skraćeno od "Matrix Laboratory", revolucionirao je pristup numeričkom računanju i modeliranju kada je prvi put predstavljen u kasnim 1970-im godinama. Razvio ga je Cleve Moler, profesor automatskog upravljanja i računarstva. Matlab je prvobitno bio zamišljen

kao alat za njegove studente, koji bi im pomogao u pristupu paketima za matrične proračune bez potrebe za učenjem programskog jezika FORTRAN. Od tada, Matlab se razvio u moćan alat koji se koristi u inženjerstvu, znanosti, ekonomiji, i mnogim drugim područjima za izvođenje složenih matematičkih izračuna, obrade podataka, simulacije i vizualizacije.



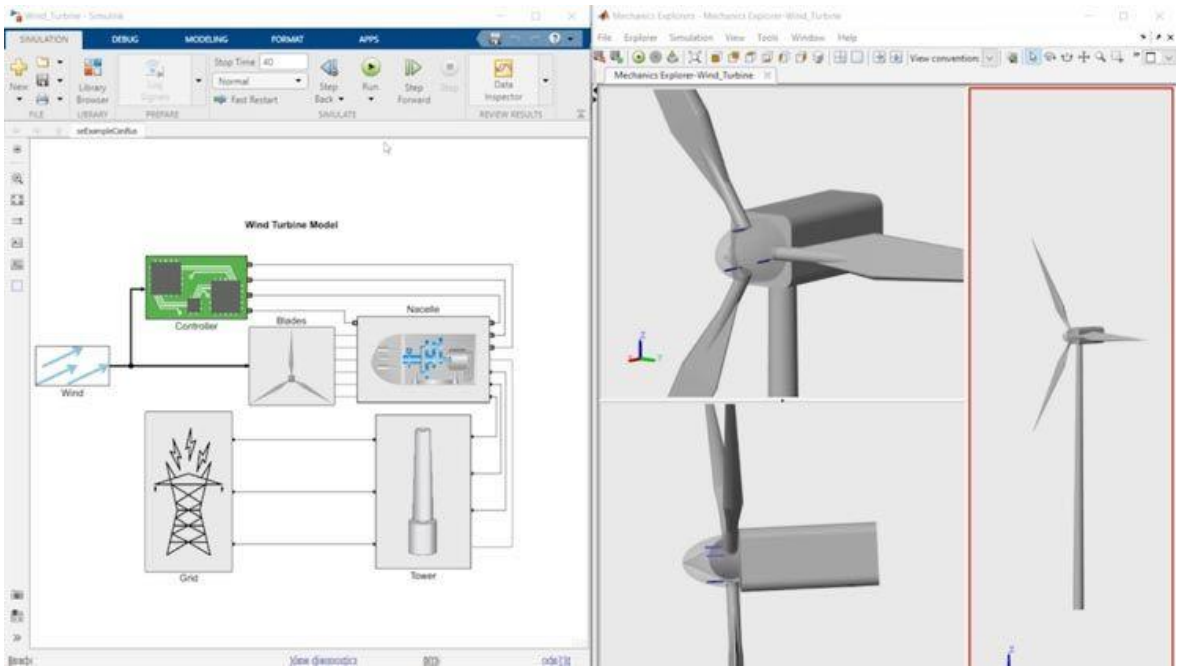
Slika 0.2 Fotografija Clevea Morela

Važnost Matlab/Simulinka u kontekstu mehatroničkih sustava je izrazito velika. Ovaj softver pruža integrirano okruženje koje omogućava multidisciplinarni pristup potreban za projektiranje mehatroničkih sustava. Kroz kombinaciju analitičkog računanja, modeliranja sustava, simulacije i vizualizacije, Matlab/Simulink omogućuje inženjerima da razvijaju, analiziraju i testiraju složene sustave prije nego što se uloži u izradu fizičkih prototipova. Ovaj proces ne samo da štedi vrijeme i resurse, već i omogućava bolje razumijevanje interakcija između različitih komponenti sustava, što je ključno u mehatronici.

Posebno značajna je sposobnost Matlab/Simulinka da se nosi s modeliranjem i simulacijom dinamičkih sustava, što je ključno za mehatroničke sustave koji često uključuju složene pokrete i interakcije između mehaničkih i elektroničkih komponenata. Simulink, kao dodatak za Matlab, pruža grafičko okruženje za modeliranje i simulaciju blok-dijagrama, što omogućava intuitivnije i efikasnije modeliranje složenih sustava.

Kroz godine, Matlab i Simulink su se kontinuirano razvijali, dodavajući nove značajke i alate koji odgovaraju rastućim potrebama industrije i akademije. Danas, ovi alati ne samo da podržavaju osnovne aspekte mehatroničkog dizajna, već se također koriste u naprednim

područjima poput umjetne inteligencije, strojnog učenja i autonomnih sustava, što ih čini nezaobilaznim u modernom inženjerskom obrazovanju i praksi.



Slika 0.3 Primjer sustava u Simulink-u

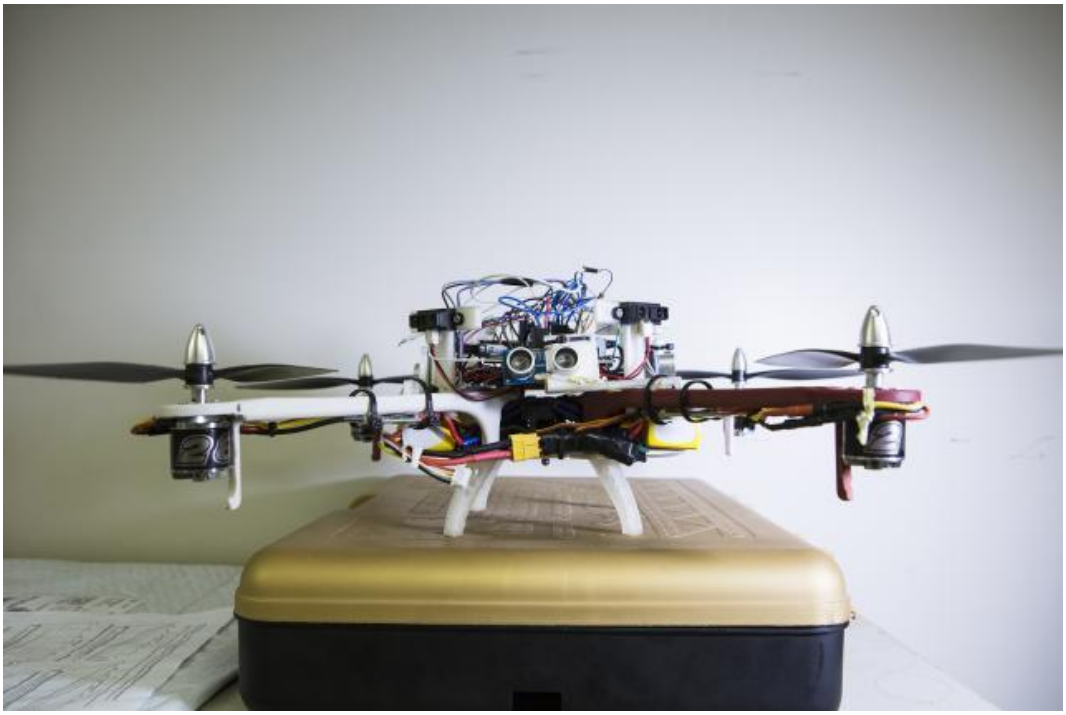
U ovom radu će detaljnije biti analizirani dodaci u Matlab-u i Simulink-u koji su potrebni za projektiranje mehatroničkih sustava. Kao ogledni primjer takvog sustava ćemo koristiti inverzno njihalo.

1. Mehanički sustavi i mehatronika: Pregled i izazovi

Ovaj odjeljak pruža temeljit uvid u osnovne koncepte mehaničkih sustava i mehatronike, te naglašava ključne izazove s kojima se suočava ovo područje, što postavlja scenu za daljnje istraživanje Matlab/Simulinka kao alata za rješavanje ovih izazova.

1.1. Definicija mehaničkih sustava i mehatronike

Mehanički sustavi su temelj tradicionalnog inženjerstva i obuhvaćaju elemente poput struktura, mehanizama, kinematike i dinamike. Ovi sustavi su ključni u stvaranju fizičkih pokreta ili snage, primjenjujući osnovne mehaničke principe i materijale. S druge strane, mehatronika predstavlja evolucijski korak u inženjerstvu koji nadilazi tradicionalne mehaničke sustave integracijom elektronike, računalnog inženjerstva i automatike. U mehatronici, mehanički dizajn se ne razmatra izolirano, već kao dio integriranog sistema koji uključuje senzore, aktuatore, i upravljačke algoritme. Ova sinergija između mehaničkih i elektroničkih komponenti dovodi do stvaranja inteligentnih, visoko adaptivnih sustava koji mogu reagirati na promjenjive okolnosti u stvarnom vremenu.



Slika 1.1 Dron, kao primjer mehatroničkog sustava

1.2. Izazovi u mehatronici

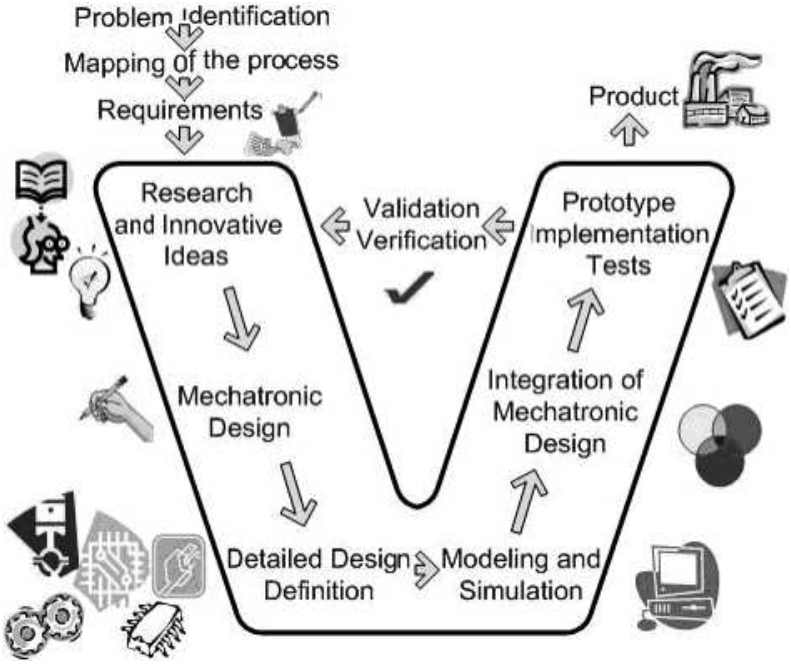
1.2.1. Integracija različitih inženjerskih disciplina

Jedan od glavnih izazova u mehatronici je integracija različitih inženjerskih disciplina. Mehanički, elektronički i računalni inženjeri moraju surađivati kako bi stvorili koherentan i funkcionalan sistem. Ovo zahtijeva razumijevanje osnovnih principa svake discipline te

izrazito dobro poznavanje jedne. Također vrlo su bitne i sposobnosti efikasne komunikacije i međusobne suradnje.

1.2.2. Složenost dizajna

Složenost dizajna je drugi značajan izazov. Mehanički sustavi u mehatronici nisu samo složeni po svojoj prirodi, već moraju biti optimizirani da efikasno komuniciraju s elektroničkim komponentama i softverom. Ovo uključuje precizno modeliranje i simulaciju kako bi se osiguralo da sve komponente rade zajedno bez problema.



Slika 1.2 Prikaz kompleksnosti modeliranja mehatroničkog sustava

1.2.3. Potreba za preciznom kontrolom i automatizacijom

Konačno, potreba za preciznom kontrolom i automatizacijom je ključna u mehatronici. Upravljački algoritmi moraju biti sofisticirani i prilagodljivi, sposobni za brzu obradu podataka i donošenje odluka u stvarnom vremenu. Ovi algoritmi moraju biti integrirani s mehaničkim i elektroničkim komponentama kako bi se osigurala visoka razina efikasnosti, pouzdanosti i sigurnosti.

Mehatronika, stoga, predstavlja područje koje zahtijeva multidisciplinarni pristup i kontinuiranu inovaciju kako bi se suočilo s ovim izazovima. Razvojem alata kao što je Matlab/Simulink, inženjeri su dobili mogućnost boljeg modeliranja, simulacije i testiranja mehatroničkih sustava, što je doprinijelo značajnom napretku u ovoj disciplini.

2. Matlab/Simulink u mehatroničkom inženjerstvu

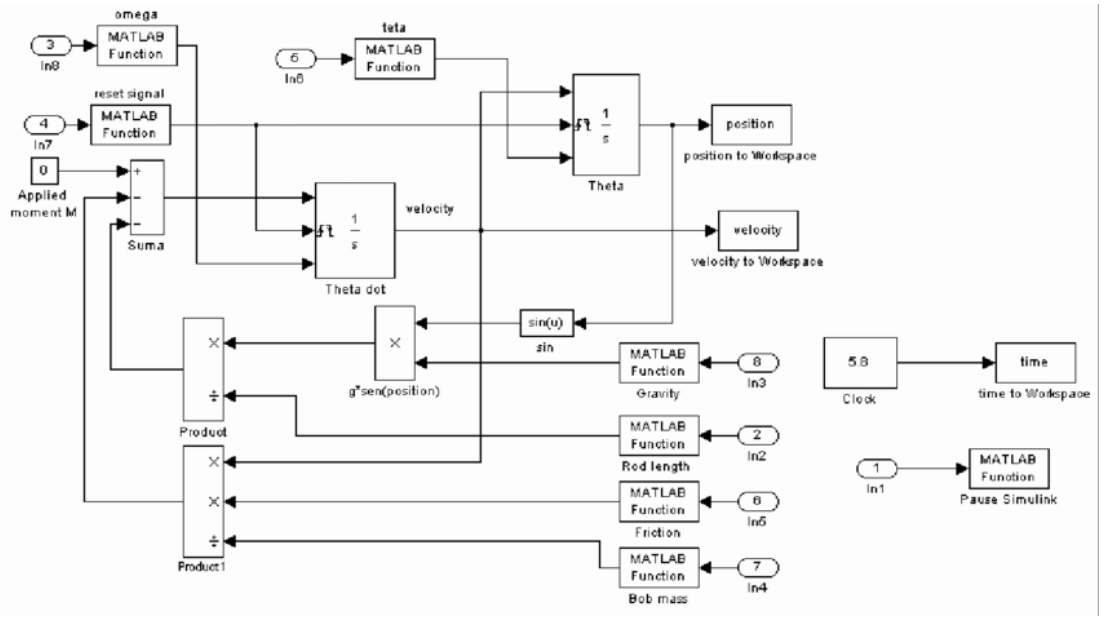
Ovaj odjeljak daje jasan pregled ključnih funkcionalnosti i primjena Matlab/Simulinka u mehatroničkom inženjerstvu, ističući kako su ovi alati integralni za razvoj, analizu i testiranje mehatroničkih sustava. Dalje, povezuje teorijske koncepte s praktičnim primjenama, pokazujući njihovu važnost u stvarnom svijetu mehatronike.

2.1. Osnovne funkcionalnosti Matlab/Simulinka

2.1.1. Opis Matlab/Simulinka

Matlab (Matrix Laboratory) je visoko specijalizirano računalno okruženje i programski jezik koji se koristi za numeričku analizu, vizualizaciju, i razvoj algoritama. Njegova primarna snaga leži u sposobnosti obrade velikih skupova podataka i matrica, što je ključno u inženjerskim i znanstvenim primjenama.

Simulink, s druge strane, je dodatak za Matlab koji pruža grafičko okruženje za modeliranje, simulaciju, i analizu dinamičkih sustava. Simulink omogućuje korisnicima da vizualno konstruiraju kompleksne modele koristeći blok-dijagrame, što olakšava razumijevanje i implementaciju sustava.



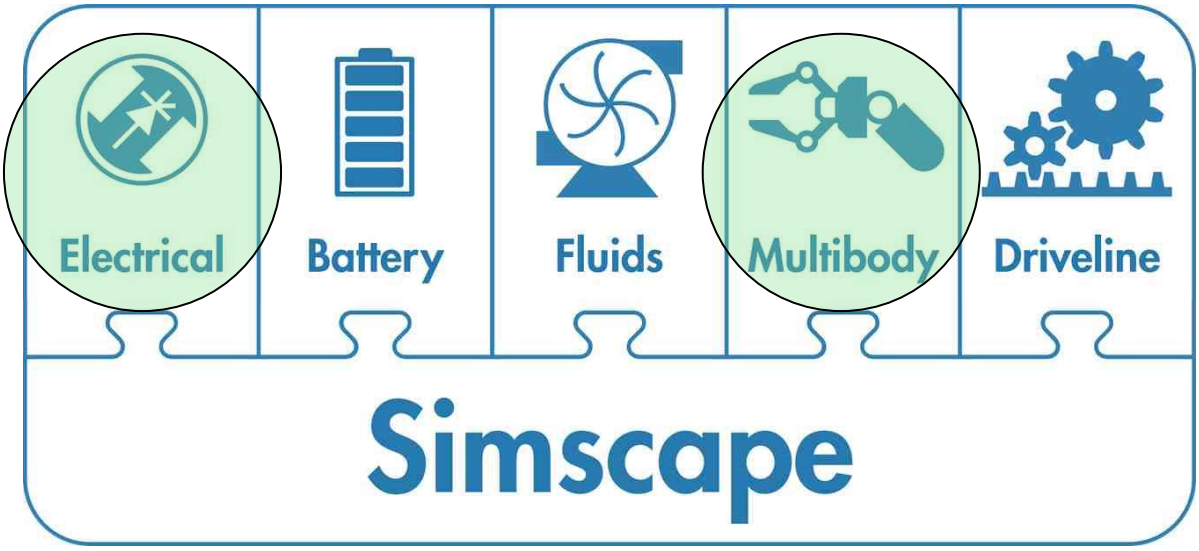
Slika 2.1 Primjer blok-dijagrama u Simulink-u

2.1.2. Specifični alati i značajke za mehatroniku

Simscape

Simscape je ključan alat unutar Matlab/Simulinka koji omogućava modeliranje i simulaciju fizičkih sustava unutar mehatronike. Simscape pruža intenzivno grafičko okruženje za simulaciju fizičkih sustava koji se temelje na komponentama, a ne na matematičkim modelima. Simscape omogućuje korisnicima da modeliraju fizičke sustave koji uključuju mehaničke, električne, hidrauličke i pneumatske komponente, sve unutar jednog modela. Ovo omogućuje preciznije i realističnije simulacije mehatroničkih sustava gdje su interakcije

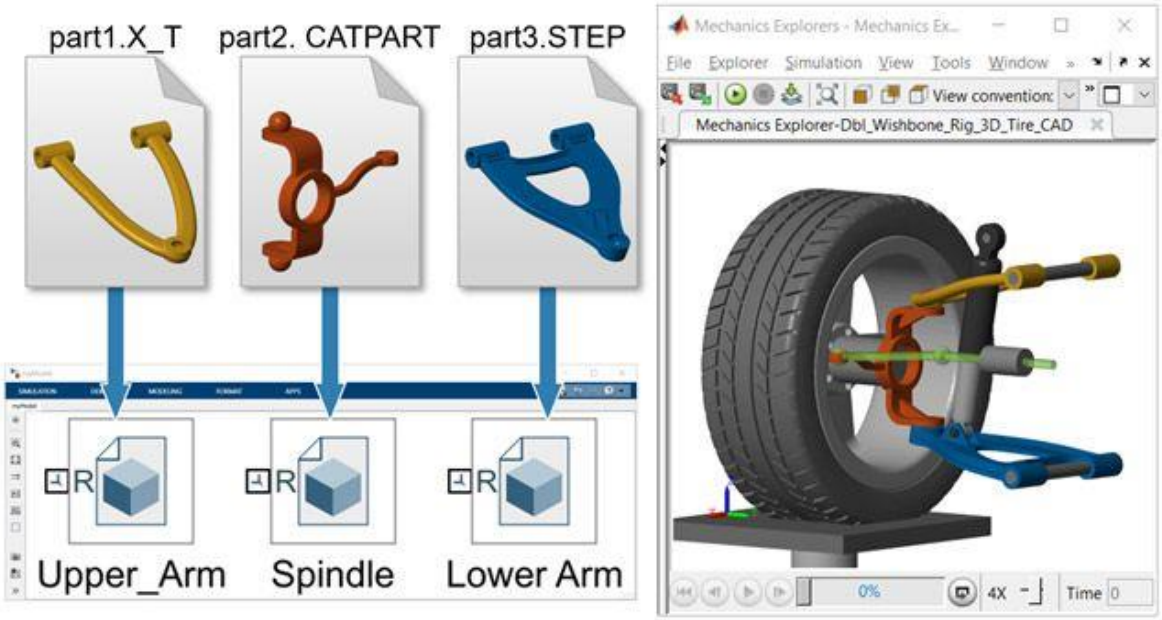
između različitih vrsta fizičkih domena ključne. Fizičke mreže i grafičko modeliranje: Simscape koristi koncept fizičkih mreža za modeliranje, gdje korisnici mogu povlačiti i spajati fizičke komponente slično kao u stvarnom svijetu. Ovo grafičko modeliranje olakšava razumijevanje i izradu kompleksnih mehatroničkih sustava. Integracija s ostalim Simulink alatima: Simscape se lako integrira s ostalim alatima unutar Matlab/Simulink okruženja, što omogućuje detaljnu analizu i simulaciju cjelokupnih mehatroničkih sustava, uključujući i njihove upravljačke algoritme. Koristili smo dva specijalizirana dijela Simscape-a: Multibody i Electrical.



Slika 2.2 Korištene komponente Simscape-a

Simscape Multibody

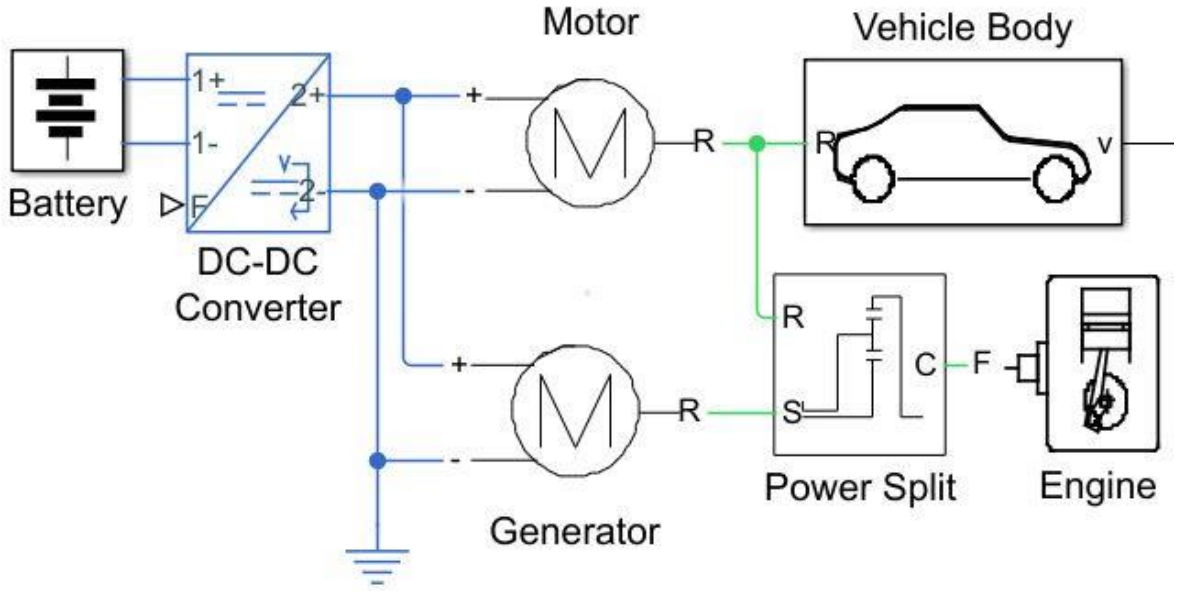
Simscape Multibody je alat unutar Matlab/Simulinka koji omogućava precizno modeliranje i analizu mehaničkih sustava u 3D prostoru. Alat koristi fizički temeljen pristup za simulaciju dinamičkog ponašanja mehaničkih sistema, što uključuje kinematiku i dinamiku. Korisnici mogu stvoriti detaljne 3D modele mehaničkih sustava, uključujući tijela, zglobove, opruge, prigušivače i druge mehaničke elemente. Ovi modeli se mogu vizualizirati i analizirati unutar Simulinka. Simscape Multibody automatski generira jednadžbe gibanja za mehaničke sustave, što uklanja potrebu za ručnim izračunavanjem kompleksnih matematičkih formulacija. Modeli stvoreni u Simscape Multibody-u kompatibilni su s ostalim Simulink blokovima, omogućavajući integraciju mehaničkog modela s elektroničkim kontrolnim sustavima i algoritmima.



Slika 2.3 Shema sustava u Simscape Multibody-u

Simscape Electrical

Simscape Electrical je također ključan alat unutar Matlab/Simulinka, dizajniran za modeliranje, simulaciju i analizu elektroničkih i elektromehaničkih sustava. Simscape Electrical nudi širok spektar modela elektroničkih komponenti kao što su tranzistori, diode, kondenzatori, otpornici i integrirana kola. Ovo omogućava korisnicima da izgrade realistične modele elektroničkih sklopova. Korisnici mogu simulirati kako elektronički sustavi reagiraju na različite uvjete i signale. Simulacije mogu uključivati analizu vremenskog odziva, frekvencijske odzive, i termalne efekte. Važan aspekt Simscape Electrical-a je njegova sposobnost da se integrira s mehaničkim modelima iz Simscape Multibody-a, omogućujući simulaciju interakcija između mehaničkih i elektroničkih sustava, što je ključno u mehatronici.



Slika 2.4 Primjer sustava u Simscape Electrical-u

Control System Toolbox

Control System Toolbox pruža set alata unutar Matlab okruženja za analizu, dizajn i simulaciju kontrolnih sistema. Ovaj toolbox je iznimno koristan za inženjere u mehatronici, posebno za razvoj i testiranje upravljačkih strategija. Control System Toolbox nudi razne funkcije za modeliranje i analizu linearnih kontrolnih sustava. Inženjeri mogu koristiti ove alate za stvaranje Bodeovih dijagrama, Nyquistovih dijagrama, i root locus plotova, što je ključno za razumijevanje ponašanja kontrolnih sistema u frekvencijskom domenu. Alat nudi i opcije za automatsko tuniranje kontrolera, kao što su PID kontroleri, što omogućava inženjerima da brzo i efikasno optimiziraju performanse svojih kontrolnih sustava. Jedna od najvećih prednosti Control System Toolboxa je njegova integracija s Matlab/Simulink okruženjem, što omogućava lako testiranje i implementaciju dizajniranih kontrolnih sistema unutar složenijih mehatroničkih modela.

2.2. Primjena Matlab/Simulinka u mehatronici

2.2.1. Modeliranje i simulacija mehatroničkih sustava

Matlab/Simulink nudi mogućnost stvaranja detaljnih modela mehatroničkih sustava. Korisnici mogu simulirati kako će se sustav ponašati u različitim uvjetima, što pomaže u identifikaciji potencijalnih problema prije izrade fizičkog prototipa.

2.2.2. Razvoj upravljačkih algoritama

Upravljački algoritmi su srž svakog mehatroničkog sustava. Matlab/Simulink omogućuje razvoj složenih algoritama za automatizaciju i kontrolu, uključujući adaptivno upravljanje, PID kontrolere, i sustave temeljene na logici. Korisnici mogu razvijati algoritme, testirati ih u simulacijama, i čak generirati kod koji se može implementirati na stvarnim hardverskim platformama.

2.2.3. Integracija i testiranje

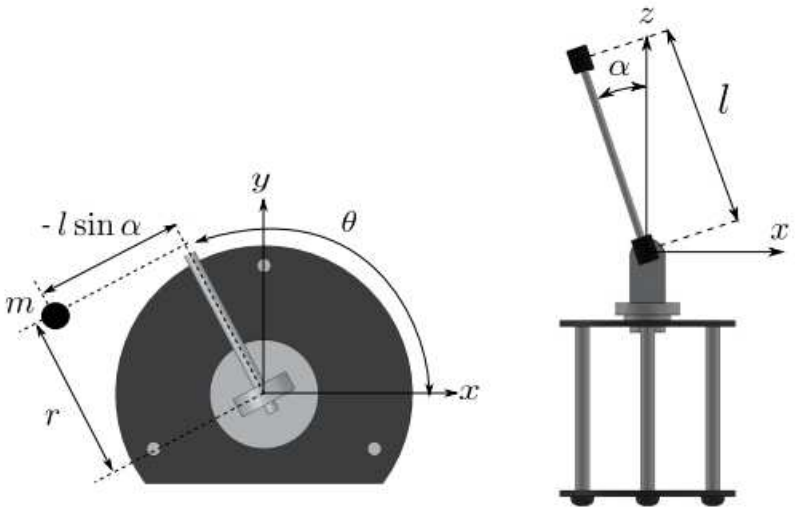
Integracija mehaničkih, elektroničkih i kontrolnih sustava je ključna u mehatronici. Matlab/Simulink olakšava ovaj proces omogućujući inženjerima da testiraju kako će različite komponente mehatroničkog sustava međusobno surađivati, identificirajući potencijalne probleme interoperabilnosti.

3. Studija slučaja: Laboratorijski postav inverznog njihala

Ulogu Matlab/Simulink-a u modeliranju mehatroničkih sustava pokazat ćemo na primjeru rotacijskog inverznog njihala.

3.1. Matematički model rotacijskog inverznog njihala

Rotacijsko inverzno njihalo se sastoji se od ruke njihala (duljine r i momenta tromosti Jr) koja rotira u horizontalnoj ravnini. Na tu ruku je ovješen štap. Kut ruke u odnosu na početni položaj označen je s θ, dok je kut njihala u odnosu na ravnotežni položaj označen s α. Pretpostavlja se pozitivan smjer rotacije ruke i njihala u smjeru obrnuto od kazaljke na satu te da pozitivan moment motora uzrokuje pozitivan smjer rotacije ruke.



Slika 3.1 Prikaz rotacijskog inverznog njihala

Potencijalna gravitacijska energija:

- $Ep = mgl\cos\alpha$

Kinetička energija razlaže se na translacijsku i rotacijsku komponentu. Kinetička energija ruke opisana je s:

- $Ek1 = \frac{1}{2}\, Jr\, \dot{\theta}^2$

Kinetička energija štapa opisana je s:

- $Ek2 = \frac{1}{2} Jcm\, \dot{\alpha}^2 + \frac{1}{2} Jcm\, sin^2\, \alpha\, \dot{\theta}^2 + \frac{1}{2} mv^2$

Pozicija centra mase štapa:

- $x = r\cos 0 + l\sin\alpha\ \sin\theta,$
- $y = \ r\sin\theta - l\sin\alpha\cos\theta,$
- $z = l\cos\alpha$

Komponente brzine centra mase štapa:

- $\dot{x} = -r\sin\theta\,\dot{\theta} + l\cos\alpha\sin\theta\,\dot{\alpha} + l\sin\alpha\cos\theta\,\dot{\theta}$
- $\dot{y} = r\cos\theta\,\dot{\theta} + l\sin\alpha\sin\theta\,\dot{\theta} - l\cos\alpha\cos\theta\,\dot{\alpha}$
- $\dot{z} = -l\sin\alpha\,\dot{\alpha}$

Kvadrat brzine centra mase štapa:

- $v^2 = \dot{x}^2 + \dot{y}^2 + \dot{z}^2$

Ukupna kinetička energija sustava (zbroj kinetičkih energija ruke i štapa):

- $Ek = \frac{1}{2}(J1 + J2sin^2\alpha)\dot{\theta}^2 + \frac{1}{2}J2\dot{\alpha}^2 - m\dot{\alpha}\dot{\theta}lr\cos\alpha$
- $J1 = Jr + mr^2$
- $J2 = Jcm + ml^2$

Tablica 3.1 Parametri sustava

simbol	opis	vrijednost	mjerna jedinica
m	masa njihala	0.107	kg
r	duljina ruke	0.09	m
l	duljina štapa	0.17	m
Jcm	moment inercije štapa	$3.5 \cdot 10^{-6}$	Kg m^2
Jr	moment inercije ruke	$4.6 \cdot 10^{-3}$	Kg m^2
g	gravitacijska konstanta	9.81	ms^2

Matematički model njihala:

- $(J_1 + J_2\sin^2(\alpha))\ddot{\theta} + J_2\sin(2\alpha)\,\dot{\theta}\dot{\alpha} - mlr\cos(\alpha)\,\ddot{\alpha} + mlr\sin(\alpha)\,\dot{\alpha}^2 = \tau$
- $-mlr\cos(\alpha)\,\ddot{\theta} + J_2\ddot{\alpha} - \frac{1}{2}J_2\sin(2\alpha)\,\dot{\theta}^2 - mgl\sin(\alpha) = 0$

3.1.1. Detaljan opis postava

Laboratorijski postav rotacijskog inverznog njihala sastoji se od nekoliko ključnih komponenti koje omogućuju precizno mjerenje i kontrolu dinamike sustava. Glavna komponenta je ruka njihala, koja rotira u horizontalnoj ravnini oko osi pričvršćene na bazu. Duljina ruke je *r*, a moment inercije ruke označen je s *Jr*. Na kraj ruke pričvršćen je štap duljine *l* i mase *m*, koji slobodno visi i može se zakretati u odnosu na ruku. Kut rotacije ruke u odnosu na početni položaj označen je s θ, dok je kut zakretanja štapa u odnosu na vertikalni ravnotežni položaj označen s α.

Sustav uključuje senzore za mjerenje kutova θ i α te njihove derivacije (brzine zakretanja). Ovi senzori su povezani s kontrolnom jedinicom koja obrađuje podatke i generira upravljačke signale za aktuator, koji primjenjuje moment τ na ruku njihala. Kontrolna jedinica koristi razvijene algoritme za održavanje stabilnosti sustava, što uključuje izračunavanje potrebnih korekcija u realnom vremenu.

Za potrebe modeliranja i simulacije, razvijen je matematički model sustava koji uključuje jednadžbe gibanja opisane u poglavlju 3.1. Ovaj model omogućuje detaljnu analizu dinamike sustava te razvoj i testiranje upravljačkih algoritama prije implementacije na fizičkom sustavu. Modeliranje se provodi u Matlab/Simulink okruženju, koristeći Simscape Multibody i druge specifične alate za simulaciju mehaničkih i elektroničkih komponenti.

3.2. Potreba Matlab/Simulinka u studiji slučaja

3.2.1. Modeliranje i simulacija

Matlab/Simulink pruža sveobuhvatno okruženje za modeliranje i simulaciju dinamike obrnutog njihala. Pomoću Simscape Multibody alata, moguće je kreirati precizan 3D model sustava koji uključuje sve mehaničke komponente i njihove međusobne interakcije. Modeliranje u Simscape Multibody omogućuje vizualizaciju kretanja sustava, što pomaže u razumijevanju složenih dinamičkih interakcija.

Simulink omogućuje stvaranje blok-dijagrama koji predstavljaju matematičke jednadžbe gibanja sustava. Ovi blok-dijagrami olakšavaju simulaciju različitih scenarija i analiza dinamike sustava pod različitim uvjetima. Na primjer, simulacija može uključivati promjene u momentu τ, različite početne uvjete za kutove θ i α, te analizu odziva sustava na te promjene. Simulacije pomažu identificirati potencijalne probleme u dizajnu i optimizirati performanse sustava prije implementacije u fizički svijet.

3.2.2. Razvoj i testiranje upravljačkih algoritama

Razvoj upravljačkih algoritama za inverzno njihalo ključan je za održavanje stabilnosti sustava. Matlab/Simulink omogućuje razvoj složenih upravljačkih algoritama koji se mogu testirati u simulacijskom okruženju prije implementacije na stvarnom hardveru. Algoritmi za kontrolu mogu uključivati PID kontrolere, adaptivne kontrolere i algoritme temeljene na logici.

Simulink pruža alate za automatsko tuniranje kontrolera, što omogućuje brzo i efikasno optimiziranje parametara upravljačkih algoritama. Tijekom testiranja, algoritmi se mogu iterativno prilagođavati kako bi se postigla optimalna stabilnost i performanse sustava. Testiranje u simulacijskom okruženju omogućuje detaljnu analizu odziva sustava na upravljačke signale i identifikaciju mogućih problema prije fizičke implementacije.

3.2.3. Implementacija i validacija

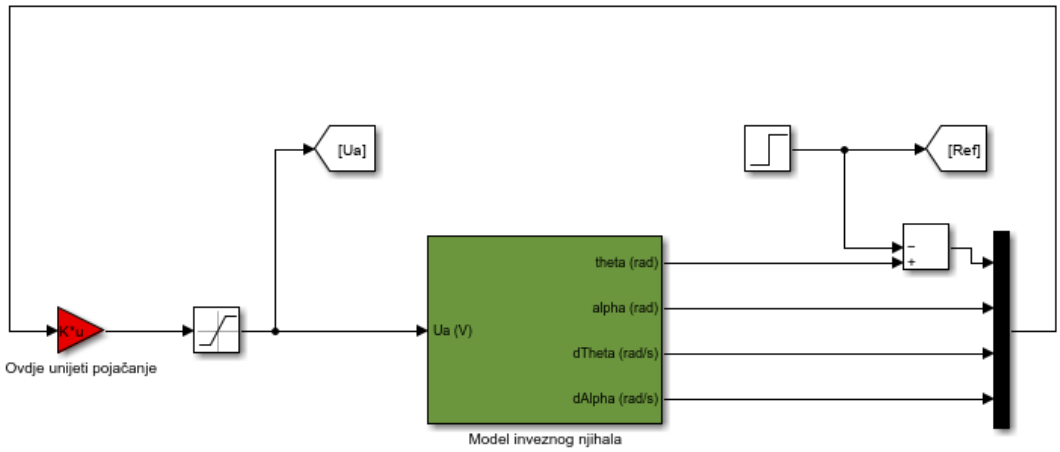
Nakon što su modeli i upravljački algoritmi razvijeni i testirani u Matlab/Simulink okruženju, sljedeći korak je implementacija na fizičku hardversku platformu. Matlab/Simulink omogućuje generiranje upravljačkog koda koji se može direktno implementirati na ugradbene računalne platforme, poput mikroprocesora i FPGA.

Proces implementacije uključuje prevođenje simulacijskih modela i algoritama u izvršni kod, koji se zatim prenosi na kontrolnu jedinicu sustava inverznog njihala. Tijekom ovog procesa, moguće je naići na izazove poput vremenskih kašnjenja u sustavu, ograničenja procesorske snage i drugih hardverskih ograničenja. Matlab/Simulink pruža alate za rješavanje ovih problema kroz optimizaciju koda i prilagodbu algoritama specifičnim hardverskim zahtjevima.

Validacija implementiranog sustava provodi se kroz eksperimentalno testiranje na fizičkom postavu. Ovaj korak uključuje usporedbu rezultata dobivenih simulacijama s stvarnim ponašanjem sustava kako bi se osigurala točnost modela i učinkovitost upravljačkih algoritama. Validacija također pomaže u identifikaciji i ispravljanju bilo kakvih odstupanja između simulacijskog modela i stvarnog sustava, čime se osigurava pouzdanost i stabilnost konačnog proizvoda.

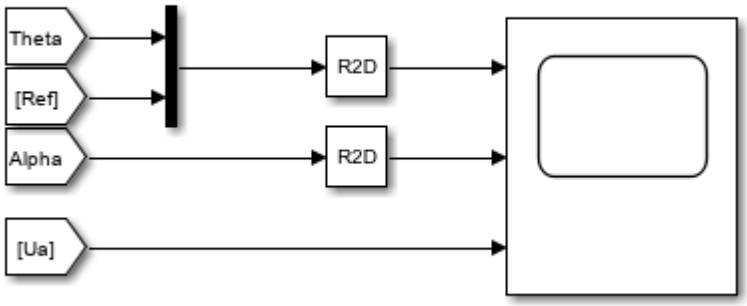
4. Analiza simulacije rotacijskog inverznog njihala

4.1. Simulink model rotacijskog inverznog njihala



Slika 4.1 Simulink model rotacijskog inverznog njihala

Na slici je prikazan blokovski Simulink model inverznog njihala u MATLAB-u. Zeleni blok predstavlja matematički model sustava kojim je opisano ponašanje njihala. Crveni blok se koristi za pojačanje, tj. povećanje ili smanjenje ulaznog signala što je važno za kontrolu sustava. Blok s [Ref] označava referentnu (željenu) vrijednost kuta koju ćemo uspoređivati s izlazom theta.



Slika 4.2 Blok modela koji služi za grafički prikaz rezultata

Blok s ekranom i ulazima vizualizira rezultate simulacije kako bi lakše mogli analizirati ponašanje sustava u odnosu na zadane parametre.

4.2. Izračun optimalnog pojačanja K

Optimalno pojačanje K ključno je za stabilizaciju dinamičkih sustava poput inverznog njihala. Sustav koristi povratnu vezu kako bi kontrolirao ponašanje inverznog njihala. Cilj

je što preciznije izračunati optimalno pojačanje K kako bi se minimizirale oscilacije i osiguralo stabilno ponašanje sustava.

4.2.1. Linearno-kvadratni regulator (LQR)

Optimalno pojačanje K često se izračunava pomoću **LQR metode** (Linear Quadratic Regulator). LQR je algoritam za optimizaciju povratne veze stanja, koji minimizira kvadratnu funkciju troškova. Ova metoda se koristi zato što omogućuje optimizaciju performansi dinamičkih sustava, uz minimalizaciju energije sustava i penaliziranje velikih odstupanja stanja od željenih vrijednosti.

- Funkcija troškova J za LQR ima oblik:

$$J = \int_0^\infty \left(x(t)^T Q x(t) + u(t)^T R u(t)\right) dt$$

Gdje:

- x predstavlja vektor stanja (npr. kutovi θ i α, te njihove brzine θ˙ i α˙,
- u predstavlja upravljački signal (u ovom slučaju napon Ua),
- Q je matrica koja penalizira odstupanja stanja od željenih vrijednosti,
- R je matrica koja penalizira prevelike upravljačke akcije.

Cilj LQR optimizacije je izračunati K koji minimizira ovu funkciju troškova. Rezultat je regulator koji može balansirati između preciznosti regulacije stanja i minimalizacije energije potrebne za upravljanje.

LQR se koristi jer osigurava:

1. **Stabilnost**: Povratna veza s optimalnim K stabilizira sustav.
2. **Minimizaciju energije**: Funkcija troškova osigurava da se prevelike upravljačke akcije penaliziraju, čime se sustav optimizira tako da koristi što manje energije.
3. **Rješavanje kompromisa**: LQR omogućuje podešavanje kompromisa između strogih zahtjeva na regulaciju stanja i ograničenja u naponu Ua.

Upravo zbog toga optimalno pojačanje K predstavlja ključnu komponentu za pravilno upravljanje sustavom inverznog njihala, gdje je potrebna izuzetno brza i precizna kontrola kako bi se osiguralo održavanje ravnoteže sustava.

U našem se modelu svi potrebni podaci nalaze u datoteci "**parametri.m**" (priložena uz rad). Prema podacima iz datoteke parametri.m, tvoj sustav uključuje različite parametre koji utječu na dinamiku inverznog njihala, kao što su:

- Otpornost motora Ra
- Brzina praznog hoda motora nn0,
- Prijenosni omjer Kg
- Parametri planetarnog zupčanika i njihala Bp, mp, Lp, Jp.

Ovi parametri direktno utječu na dinamiku sustava te će biti korišteni prilikom računanja optimalnog pojačanja K. Optimalno K mora uzeti u obzir inercije i otporne sile koje uzrokuju trenje (npr. Bp), te težinu i dužinu njihala (npr. mp, Lp) kako bi sustav mogao reagirati brzo, ali i stabilno.

Datoteka parametri.m nam pomoću lqr funkcije u Matlab-u i naših parametara izračunava K koji koristimo u modelu i simulacijama.

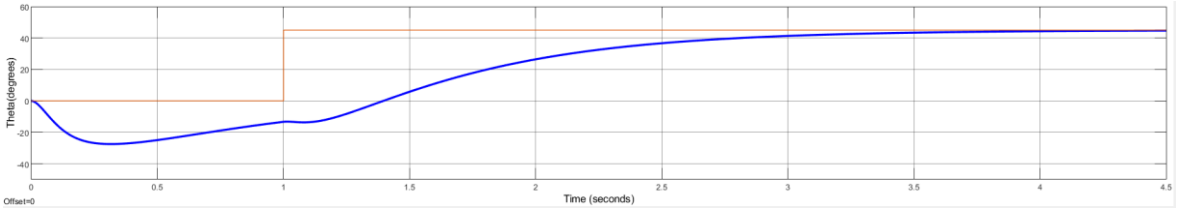
4.3. Simulacija sustava stabilizacije inverznog njihala

Prije pokretanja gore navedenog modela moramo odrediti pojačanje K*u. Prethodnim izračunom smo dobili da bi sustav trebao biti u ravnoteži za:

- K*u = -[-0.8944 27.5038 -1.8420 3.2782]

Pokretanjem simulacije želimo potvrditi naš izračun i očekivanja. Rezultati su najjasnije vidljivi pomoću grafova koje smo ugradili u naš blok model.

(Grafovi su nepregledni zbog svog "duguljastog" oblika. Bit će dodatno priloženi kao slike uz ovaj rad)

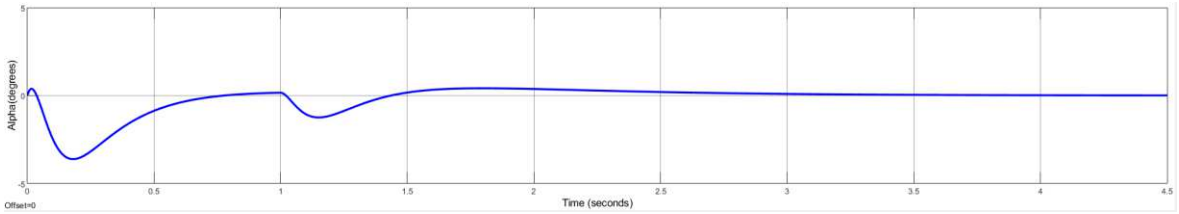


Slika 4.3 Graf ovisnosti kuta ruke o vremenu

Prvi graf prikazuje kako se kut θ (kut zakreta ruke) njihala mijenja tijekom vremena u odnosu na zadanu referentnu vrijednost. Narančastom bojom prikazana je referentna vrijednost koja skače na 45 stupnjeva u prvoj sekundi. Ona se koristi za procjenu dinamike i stabilnosti sustava. Plavom bojom prikazan je stvarni odgovor sustava.

Ključne točke analize:

1. **Početna reakcija**: Na početku, kut θ započinje od nule, što znači da je sustav inicijalno u ravnoteži.
2. **Prva oscilacija**: Unutar prvih 0,5 sekundi, kut θ pada ispod nule. Ova oscilacija je reakcija sustava na promjenu koja se događa jer sustav na početku nije u savršenoj ravnoteži. Štap počinje padati pa se ruka prilagođava.
3. Impulsna reakcija: Pri skoku referentne vrijednosti (oko prve sekunde), dolazi do pada kuta θ, što je posljedica pokušaja sustava da brzo prilagodi kut θ na impuls.
4. **Daljnje oscilacije**: Nakon prve sekunde, sustav prolazi kroz još nekoliko oscilacija dok pokušava stabilizirati kut θ. Ove oscilacije su manje intenzivne nego početne oscilacije.
5. **Stabilizacija**: Nakon približno 2,5 sekunde, oscilacije se smanjuju, a kut θ se postupno stabilizira oko referentne vrijednosti od 45 stupnjeva. Do kraja vremena simulacije, kut θ ostaje stabilan i ne pokazuje značajne oscilacije.

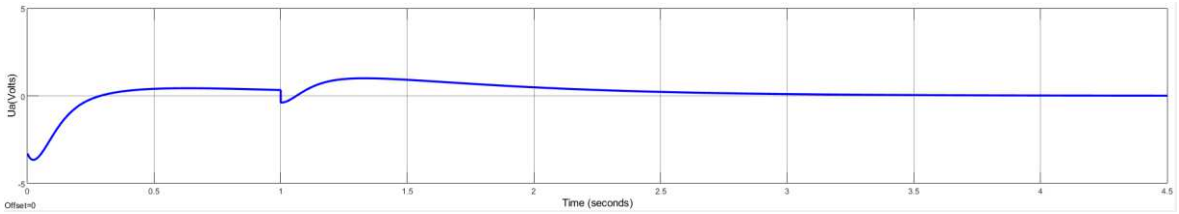


Slika 4.4 Graf ovisnosti kuta štapa o vremenu

Drugi graf prikazuje kako se kut α (kut zakreta štapa) njihala mijenja tijekom vremena. Plava linija predstavlja stvarni odgovor sustava.

Ključne točke analize:

1. **Početna reakcija**: Na početku grafika, kut α započinje od nule, što znači da je sustav inicijalno u ravnoteži.
2. **Prva oscilacija**: Unutar prvih 0,5 sekundi, kut α raste do maksimalne vrijednosti, zatim se smanjuje ispod nule, te ponovno raste. Ova oscilacija je reakcija sustava na promjenu referentne vrijednosti θ.
3. **Impulsna reakcija**: Pri skoku referentne vrijednosti (oko prve sekunde), dolazi do naglog pada kuta α, što je posljedica pokušaja sustava da brzo prilagodi kut θ.
4. **Daljnje oscilacije**: Nakon prve sekunde, sustav prolazi kroz još nekoliko oscilacija dok pokušava stabilizirati kut α. Ove oscilacije su manje intenzivne nego početne oscilacije.
5. **Stabilizacija**: Nakon približno 2,5 sekunde, oscilacije se smanjuju, a kut α se postupno stabilizira oko nule. Do kraja vremena simulacije, kut α ostaje stabilan i ne pokazuje značajne oscilacije.



Slika 4.5 Graf ovisnosti napona o vremenu

Treći graf prikazuje kako se napon Ua (napon upravljanja sustavom) njihala mijenja tijekom vremena. Plava linija predstavlja stvarni odgovor sustava.

Ključne točke analize:

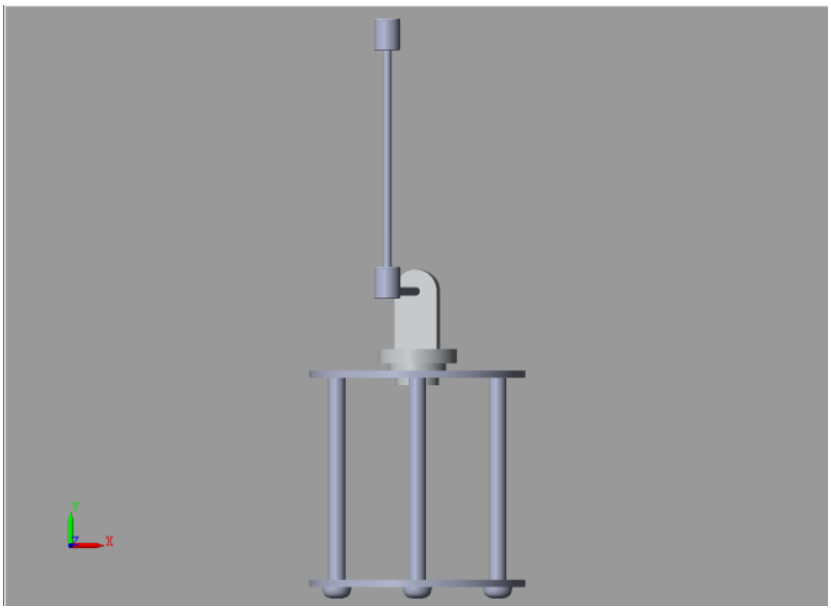
1. **Početna reakcija**: Na početku grafika, napon Ua je negativan jer kontroler pokušava ispraviti odstupanje od referentne vrijednosti ili uspostaviti ravnotežu sustava. Ovo može biti zbog inicijalnog stanja sustava koje zahtijeva korektivnu akciju.
2. **Prva oscilacija**: Unutar prvih 0,5 sekundi, napon Ua brzo raste do maksimalne vrijednosti, a zatim stagnira. Ova oscilacija je reakcija sustava na početne uvjete i inicijalno podešavanje kontrolera.
3. **Impulsna reakcija**: Pri skoku referentne vrijednosti (oko prve sekunde), napon Ua se naglo mijenja, što je pokušaj sustava da brzo prilagodi kut θ i α. Ova promjena je potrebna za postizanje nove referentne vrijednosti.

4. **Daljnje oscilacije**: Nakon prve sekunde, sustav prolazi kroz još nekoliko manjih oscilacija dok pokušava stabilizirati napon Ua. Ove oscilacije su manje intenzivne nego početne oscilacije jer sustav postepeno prigušuje dinamiku.
5. **Stabilizacija**: Nakon približno 2,5 sekunde, oscilacije se smanjuju, a napon Ua se postupno stabilizira. Do kraja vremena simulacije, napon Ua ostaje stabilan i ne pokazuje značajne oscilacije, što ukazuje na to da je sustav dosegao ravnotežu.

Zaključci:

- **Dinamika sustava**: Sustav pokazuje inicijalnu prekomjernu reakciju i oscilacije kao odgovor na promjenu referentne vrijednosti. Ovo je tipično za dinamičke sustave kada se suočavaju s naglim promjenama.
- **Stabilizacija**: Sustav se na kraju stabilizira, što znači da zadani model uspješno prigušuje oscilacije i održava sustav u ravnoteži.
- **Performanse regulatora**: Iako sustav na kraju doseže stabilnost, inicijalne oscilacije ukazuju na to da bi regulator mogao biti optimiziran za brži odgovor s manje oscilacija.

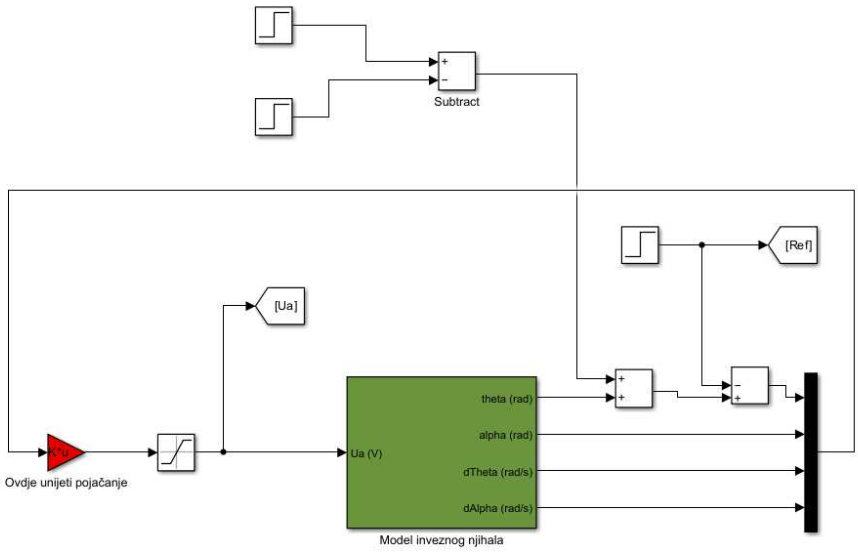
Osim grafovima, simulaciju smo mogli analizirati i pomoću 3D modela koji omogućuje vizualnu inspekciju gibanja i dinamike sustava. Mechanics Explorer omogućuje vizualizaciju modela napravljenih u Simscape Multibody-u.



Slika 4.6 Prikaz ravnotežnog stanja njihala

Slika iznad prikazuje završno ravnotežno stanje našeg modela nakon provedene simulacije. Vidimo da su odabrani parametri uspjeli održati sustav u ravnotežnom položaju čak i nakon impulsne pobude.

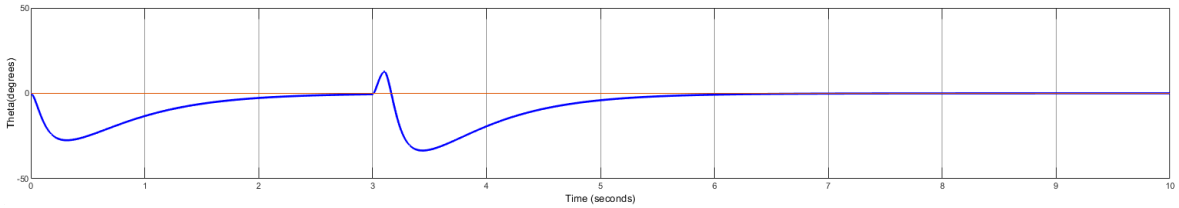
4.4. Simulacija sustava s impulsom



Slika 4.7 Simulink model inverznog njihala s dodanim impulsom

U već postojeći model dodana su dva "Step" bloka s kojima se simulira kratkotrajni trzaj na ruku njihala.

4.4.1. Simulacija stabilnog vladanja sustava



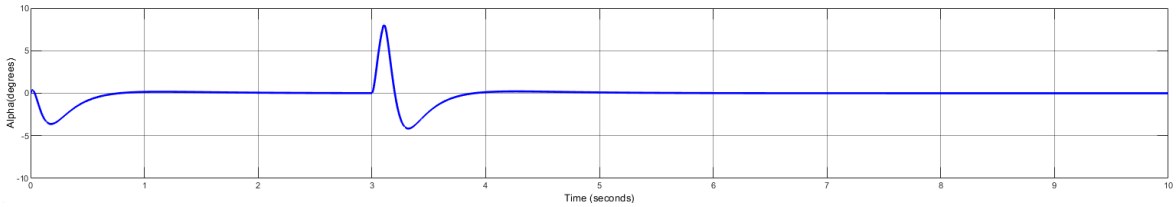
Slika 4.8 Graf ovisnosti kuta ruke o vremenu

Analizirajući graf koji prikazuje promjenu kuta θ u stupnjevima u odnosu na vrijeme, možemo identificirati nekoliko ključnih točaka.

Ključne točke analize:

1. **Početna reakcija**: Na samom početku grafa, kut θ naglo opada prema negativnim vrijednostima. To znači da je njihalo krenulo izvan ravnotežne pozicije te je sustav odmah reagirao pokušavajući korigirati ovo odstupanje.
2. **Prva oscilacija**: Do treće sekunde sustav se ujednačeno vraća prema početnom stanju.
3. **Trzaj oko 3. sekunde**: Najizraženija promjena događa se u trećoj sekundi kada dolazi do naglog pozitivnog skoka i pada vrijednosti kuta. Ovaj trzaj se događa zbog zadanog impulsa koji mijenja uvjete sustava što rezultira drastičnim pomicanjem kuta. Nakon tog pada se sustav pokušava stabilizirati.
4. **Daljnje oscilacije**: Nakon treće sekunde, pojavljuju se oscilacije koje se postupno smanjuju, kako sustav traži stabilnost. Oscilacije su blaže u usporedbi s početnim reakcijama, ali su još uvijek prisutne.

5. **Stabilizacija**: Oko 6. sekunde dolazi do potpune stabilizacije kuta θ, gdje oscilacije gotovo u potpunosti prestaju, a sustav ostaje u ravnoteži do kraja simulacije.

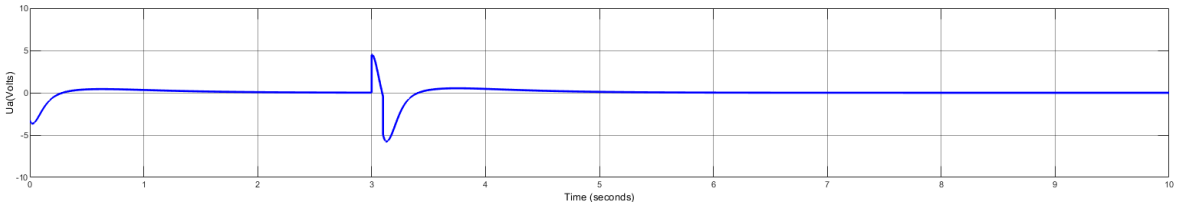


Slika 4.9 Graf ovisnosti kuta štapa o vremenu

Analizirajući graf koji prikazuje promjenu kuta α u odnosu na vrijeme, ključne točke analize su sljedeće:

Ključne točke analize:

1. **Početna reakcija**: Na početku grafa, kut α pokazuje blagu negativnu oscilaciju ispod nulte linije. Ovo sugerira da je sustav, slično kao kod kuta θ, krenuo s korektivnom akcijom kako bi stabilizirao početno odstupanje.
2. **Prva oscilacija**: Unutar prve sekunde, kut α se kreće prema traženoj vrijednosti. Sustav u ovom trenutku još uvijek traži ravnotežu.
3. **Trzaj oko 3. sekunde**: Oko treće sekunde dolazi do naglog skoka u pozitivnom smjeru, slično kao i kod kuta θ. Ovaj trzaj je povezan s promjenom referentne vrijednosti zbog impulsa koji zahtijeva brzu korektivnu akciju.
4. **Daljnje oscilacije**: Nakon treće sekunde, kut α ulazi u fazu manjih oscilacija. Ove oscilacije su karakteristične za sustav koji se prilagođava novim uvjetima i pokušava se stabilizirati.
5. **Stabilizacija**: Oko 5. sekunde, kut α se počinje stabilizirati i oscilacije postaju gotovo neprimjetne. Do kraja simulacije, vrijednost kuta ostaje stabilna, što ukazuje da je sustav postigao ravnotežu.



Slika 4.10 Graf ovisnosti napona o vremenu

Analizirajući graf koji prikazuje promjenu napona Ua u odnosu na vrijeme, ključne točke analize su sljedeće:

Ključne točke analize:

1. **Početna reakcija**: U početku vidimo nagli porast napona Ua prema pozitivnim vrijednostima. Ovo je tipičan odgovor sustava kada kontroler inicira korektivnu akciju kako bi reagirao na početno odstupanje sustava.
2. **Prva oscilacija**: Nakon ovog početnog porasta, dolazi do blage oscilacije, gdje napon Ua pokazuje blage promjene, ali se relativno brzo stabilizira, ukazujući na uspješan početak regulacije.
3. **Trzaj oko 3. sekunde**: Kao i kod prethodnih grafova, najveći skok događa se oko treće sekunde, kada dolazi do naglog opadanja napona prema negativnim

vrijednostima. Ova promjena ukazuje na korektivnu akciju sustava uslijed naglog događaja. Nakon toga dolazi do naglog povratka napona prema pozitivnim vrijednostima.

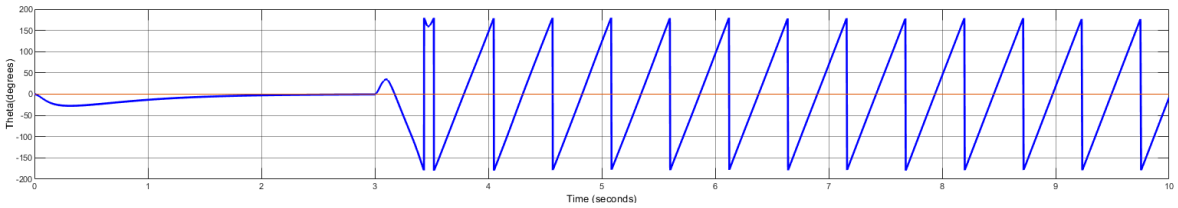
4. **Daljnje oscilacije**: Oscilacije nakon treće sekunde postupno se smiruju, ali postoji jasna oscilacija u odgovoru nakon skoka napona, što je karakterističan pokušaj sustava da se stabilizira.
5. **Stabilizacija**: Nakon otprilike 5. sekunde, napon Ua postaje stalan i stabilizira se oko nulte vrijednosti, bez značajnih oscilacija do kraja simulacije. Ovo ukazuje na to da je sustav postigao stabilnost i da više nema potrebe za daljnjim korektivnim radom.

Zaključci:

- **Dinamika sustava**: Sustav prikazuje karakteristične oscilacije na početku simulacije, pri čemu su trzaji i najveće promjene primjetne oko treće sekunde. Ova dinamika upućuje na to da sustav reagira na promjene referentne vrijednosti zbog vanjske smetnje. Nagli skokovi i promjene kod napona Ua, kutova θ i α ukazuju na snažnu početnu korektivnu akciju sustava kako bi se vratio u ravnotežu.
- **Stabilizacija**: Nakon inicijalnih oscilacija i skokova u trećoj sekundi, sustav postepeno smanjuje oscilacije u svim promatranim stanjima. Oko 5. sekunde, sve varijable stanja se stabiliziraju, što pokazuje da sustav uspješno doseže ravnotežu bez značajnih oscilacija do kraja simulacije.
- **Performanse regulatora**: Regulator uspješno stabilizira sustav nakon početnih oscilacija, no trzaji oko treće sekunde ukazuju na prostor za poboljšanje. Regulator bi mogao biti optimiziran kako bi se smanjila intenzivnost oscilacija i trzaja uslijed vanjskih utjecaja te omogućio brži i precizniji odgovor sustava bez tako naglih promjena.

4.4.2. Simulacija nestabilnog vladanja sustava

U ovom slučaju je dodatno povećana vrijednost impulsa kojom simuliramo trzaj nad sustavom. Vrijednost je dovoljno velika da sustavu nepovratno onemogući stabilizaciju.



Slika 4.11 Graf ovisnosti kuta ruke o vremenu

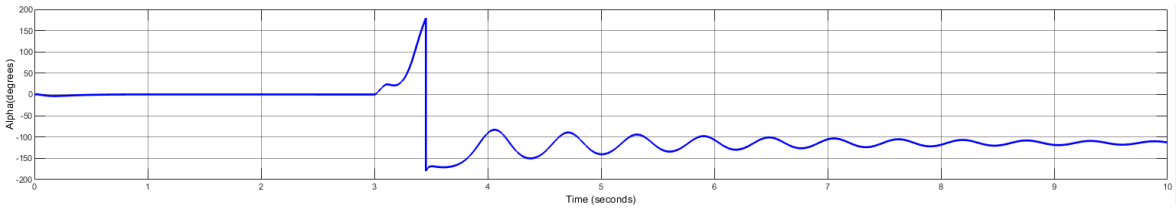
Na temelju grafičkog prikaza kuta θ u odnosu na vrijeme, možemo odmah primijetiti da je trzaj u trećoj sekundi izuzetno jak, te sustav ne uspijeva stabilizirati kut.

Ključne točke analize:

1. **Početna reakcija**: U prve 2 do 3 sekunde sustav pokazuje relativno stabilnu reakciju s blagim oscilacijama. Početno odstupanje je svedeno na minimum, a sustav se kreće prema stabilizaciji prije nego što dođe do dramatične promjene u trećoj sekundi.
2. **Trzaj oko 3. sekunde**: U trećoj sekundi dolazi do izrazito snažnog trzaja koji gura kut θ daleko izvan očekivanih vrijednosti, prema negativnim vrijednostima. Trzaj je

toliko jak da uzrokuje gotovo "eksplozivan" porast oscilacija u oba smjera, s iznimno velikim amplitudama. Dolazi do problema u regulaciji.

3. **Nekontrolirane oscilacije**: Nakon treće sekunde, umjesto da sustav smiruje oscilacije, one postaju sve jače i učestalije. Oscilacije između pozitivnih i negativnih vrijednosti kuta su vrlo velike, što sugerira da kontrolni mehanizam nije sposoban zaustaviti ili prigušiti ove oscilacije. Kut se nekontrolirano oscilira s amplitudama, što je znak neuspjeha regulatora.

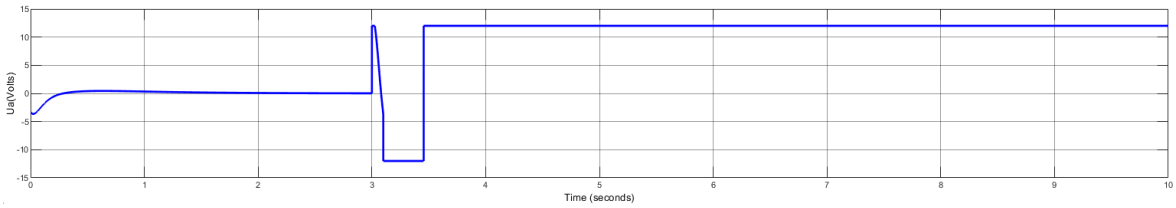


Slika 4.12 Graf ovisnosti kuta štapa o vremenu

Analizirajući graf koji prikazuje promjenu kuta α (u stupnjevima) u odnosu na vrijeme, ključne točke analize za ovu simulaciju su sljedeće:

Ključne točke analize:

1. **Početna reakcija**: U prve 2 do 3 sekunde sustav pokazuje stabilan i ujednačen odgovor s minimalnim oscilacijama. Ovo sugerira da je sustav isprva uspješno kontrolirao kut α, održavajući ga blizu referentne vrijednosti.
2. **Trzaj oko 3. sekunde**: Oko treće sekunde dolazi do izrazito snažnog trzaja. Kut α drastično raste, nakon čega dolazi do naglog pada ispod nulte linije. Ovaj nagli skok i promjena u reakciji sustava ukazuju na poremećaj i naglu promjenu u referentnoj vrijednosti, na što sustav ne uspijeva adekvatno reagirati. Iako je oscilacija manja nego kod kuta θ, ova pojava pokazuje veliki nedostatak stabilnosti.
3. **Oscilatornost odziva**: Nakon treće sekunde, kut α ne ostaje stabilan, već ulazi u seriju oscilacija. Međutim, ove oscilacije postaju manje izražene s vremenom, što sugerira da sustav pokušava stabilizirati, ali i dalje nije postigao ravnotežu jer je štap pao i ne može se podići u stabilan položaj. Sustav se nalazi u dinamičkom stanju s relativno blagim oscilacijama oko neke negativne vrijednosti između 4. i 10. sekunde.



Slika 4.13 Graf ovisnosti napona o vremenu

Analizirajući graf koji prikazuje napon Ua u odnosu na vrijeme, možemo uočiti nekoliko važnih aspekata u ponašanju sustava:

Ključne točke analize:

1. **Početna reakcija**: Na početku, napon Ua pokazuje blagi porast prema pozitivnim vrijednostima, slično kao i u prethodnim simulacijama. Sustav počinje s inicijalnom

korektivnom akcijom, ali ovaj dio simulacije pokazuje stabilno ponašanje, bez značajnih oscilacija u naponskom odgovoru.

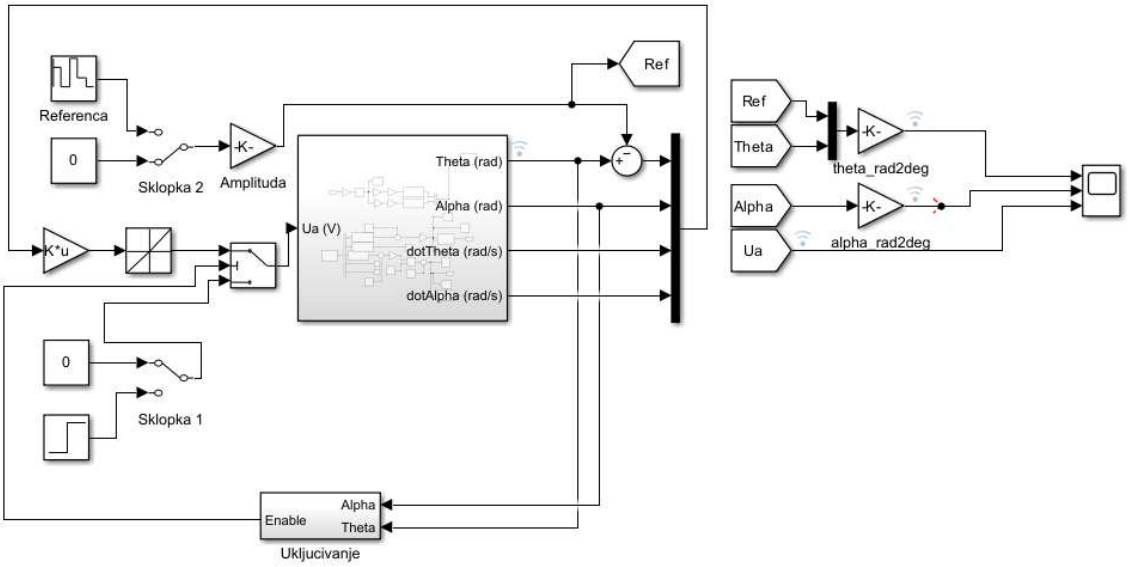
2. **Trzaj oko 3. sekunde**: Oko treće sekunde dolazi do naglog i snažnog pada napona Ua, koji pada do približno -10 volti. Ova promjena ukazuje na veliki poremećaj u sustavu, sličan kao i kod kutova θ i α. Nakon toga, napon se brzo vraća nastavlja se stabilizirati iznad pozitivnih vrijednosti (oko 10 volti). Ova dramatična promjena pokazuje da sustav u trećoj sekundi nije uspješno prigušio snažan poremećaj.
3. **Stabilizacija nakon trzaja**: Nakon što je napon dostigao stabiliziranu vrijednost iznad 10 volti, sustav više ne pokazuje nikakve oscilacije ili fluktuacije do kraja simulacije. Međutim, činjenica da se napon stabilizira na vrlo visokim pozitivnim vrijednostima sugerira da sustav više ne uspijeva vratiti napon Ua na neutralnu ili referentnu vrijednost, što ukazuje na grešku u kontrolnom procesu.

Zaključci:

- **Dinamika sustava i izražena oscilatornost u odzivu**: U ovoj simulaciji, sustav pokazuje izrazito neuspješnu kontrolu nakon treće sekunde, gdje dolazi do snažnog trzaja u svim parametrima. Ovaj trzaj je prejak i uzrokuje značajne poremećaje u sustavu. Kut θ ulazi u nekontrolirane oscilacije, dok se kut α nakon trzaja stabilizira, ali i dalje manje intenzivno oscilira oko krive vrijednosti, iako manje intenzivno. Napon Ua također doživljava dramatičan skok, nakon kojeg ostaje stabiliziran na visokoj pozitivnoj vrijednosti, što je znak nestabilnosti. Ova simulacija jasno pokazuje da sustav nije sposoban uspješno kontrolirati ovako velike poremećaje i održati stabilnost nakon početnog trzaja.
- **Performanse regulatora**: Regulator nije optimiziran za ovakve poremećaje u sustavu. Veliki trzaj u trećoj sekundi nije prigušen na odgovarajući način, što dovodi do nekontroliranih oscilacija i nestabilnosti. Potrebno je ozbiljno optimizirati regulator kako bi se smanjila osjetljivost na trzaje i poboljšala sposobnost prigušenja oscilacija.

5. Analiza postava rotacijskog inverznog njihala

5.1. Simulink model postava rotacijskog inverznog njihala



Slika 5.1 Blok dijagram Simulink modela rotacijskog inverznog njihala

Prikazani blokovski dijagram u Simulink okruženju MATLAB-a predstavlja model inverznog rotacijskog njihala. Ovaj model ćemo koristiti za povezivanje sa stvarnim hardverskim postavom inverznog njihala, koji je priključen na laptop. Aktivacijom simulacije pomoću naredbe **Run**, stvarno inverzno njihalo započinje svoj rad, pri čemu se prikupljeni podaci o kretanju, kao što su kut zakreta (θ), kut nagiba (α) i njihove derivacije brzine, bilježe u realnom vremenu.

Povezivanje Simulink modela s fizičkim sustavom omogućuje nam dvosmjernu komunikaciju između modela i stvarnog postava. Na taj način, ulazni signali generirani u MATLAB-u koriste se za upravljanje stvarnim postavom, dok se povratni podaci sa senzora na njihalu kontinuirano zapisuju i vizualiziraju u MATLAB-u. To omogućuje detaljnu analizu ponašanja sistema u stvarnom vremenu, usporedbu simuliranih i stvarnih podataka te optimizaciju kontrolnih algoritama.

Takav pristup omogućuje inženjerima i istraživačima da testiraju i validiraju različite kontrolne strategije u sigurnom simulacijskom okruženju prije implementacije na fizičkom sustavu, čime se smanjuje rizik od oštećenja opreme i poboljšava učinkovitost razvoja.

5.2. Povezivanje postava s računalom pomoću Simulink-a

Simulink je neophodan dio za povezivanjem stvarnog postava s računalom u svrhu prikupljanja podataka.

Stvarni sustav inverznog njihala povezuje se s računalom putem USB sučelja. USB kabel se koristi za prijenos podataka između mikrokontrolera na sustavu i MATLAB-a na računalu.

Na ploči mikrokontrolera, priključenoj na sustav, nalaze se senzori koji prikupljaju podatke poput kutne brzine, kuta zakreta i kuta nagiba. Ovi podaci se putem USB-a prenose u MATLAB.

Simulink se koristi za stvaranje virtualnog modela sustava, koji u stvarnom vremenu komunicira s fizičkim postavom. Model je kreiran tako da omogućuje kontrolu parametara sustava i simulaciju algoritama kontrole, koji se primjenjuju na stvarnom postavu. Također, omogućuje dvosmjernu komunikaciju između računala i stvarnog sustava: podaci o stvarnom stanju inverznog njihala šalju se u Simulink model, dok izlazni signali generirani u modelu šalju povratne informacije mikrokontroleru, čime se ostvaruje upravljanje pokretom njihala.

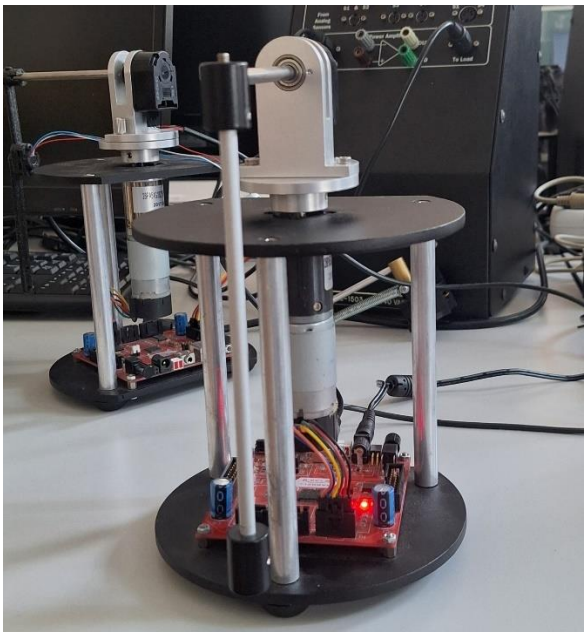
Mikrokontroler na ploči sustava upravlja motorom i obrađuje signale sa senzora. Ovi podaci se koriste za izračunavanje odgovarajućih kontrolnih signala u Simulinku. Korištenjem **Desktop Real-Time** dodatka za Simulink, omogućen je rad u stvarnom vremenu, gdje se podaci iz stvarnog sustava kontinuirano primaju i obrađuju, a upravljački algoritmi implementirani u Simulinku šalju odgovarajuće naredbe sustavu putem mikrokontrolera.

Desktop Real-Time je alat u okviru MATLAB-a i Simulink-a koji omogućuje izvođenje simulacija u stvarnom vremenu na standardnom računalu. Prilikom pokretanja s opcijom **Run in Real Time** model prolazi kroz tri faze; Build, Connect i Start. Ovaj alat se koristi u raznim inženjerskim aplikacijama, posebno u situacijama gdje je važno uskladiti simulaciju sa stvarnim sustavom, poput mehaničkih sustava, upravljačkih algoritama ili sustava s povratnim vezama, kao što je i ovaj postav inverznog njihala.

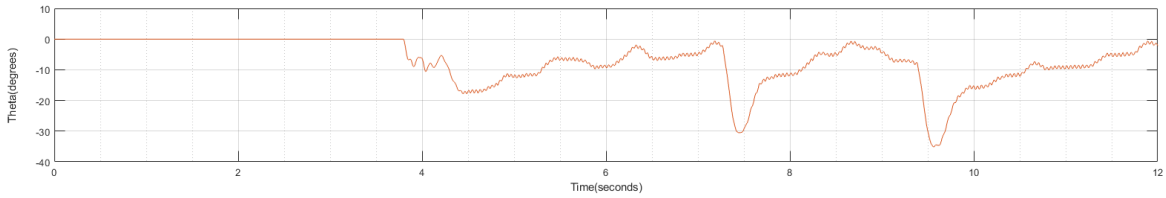
Pomoću Simulinka se rezultati rada stvarnog sustava mogu grafički prikazivati i analizirati u MATLAB-u. To uključuje podatke poput kutova, brzina i akceleracija koje senzori bilježe dok se stvarno njihalo kreće.

5.3. Analiza podataka dobivenih s postava rotacijskog inverznog njihala

U ovom poglavlju ćemo analizirati kako se stvarni sustav ponaša i usporediti sa simuliranim.



Slika 5.2 Postav rotacijskog inverznog njihala

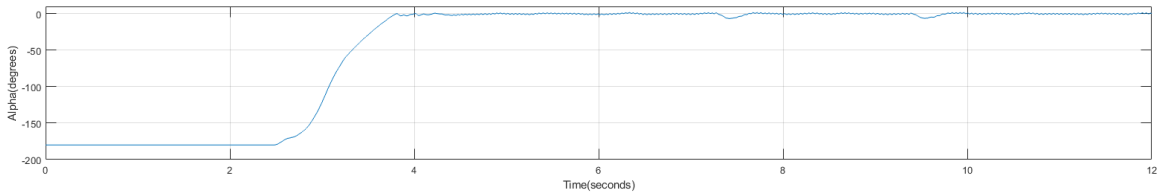


Slika 5.3 Graf ovisnosti kuta ruke o vremenu

Analizirajući graf koji prikazuje kut θ u odnosu na vrijeme za stvarni postav, možemo identificirati sljedeće ključne aspekte ponašanja sustava:

Ključne točke analize:

1. **Početna stabilnost**: U prvih 3 sekunde sustav održava stabilnost kuta θ, bez značajnih promjena. Ovo ukazuje na stabilan početak eksperimenta bez većih poremećaja ili trzaja. Vrijednost kuta je blizu 0, što je očekivano za ravnotežnu točku. Ništa se ne događa jer još nismo rukom podigli štap do položaja gdje sustav kreće sa stabiliziranjem.
2. **Trzaj prije četvrte sekunde**: Oko 4. sekunde vidimo prvi trzaj, gdje kut θ naglo pada ispod nulte linije. Ovo ukazuje na poremećaj ili promjenu referentne vrijednosti, na što sustav reagira pokušavajući se prilagoditi. Međutim, nakon prvog trzaja, kut ne ostaje stabilan, već ulazi u seriju manjih oscilacija. Ovo se krenulo događati kad smo rukom podigli štap. Sustav se tad krenuo prilagođavati te se kut prilagođavao tako da štap ostane uspravno.
3. **Oscilacije i stabilizacija**: Od 4. do 6. sekunde sustav ulazi u seriju oscilacija s relativno malim amplitudama koje postupno postaju veće, ali ne prelaze kritične granice. Ovo ponašanje ukazuje na postepeni povratak prema stabilizaciji. Iako oscilacije nisu potpuno eliminirane, sustav održava prihvatljive promjene u kutu, što ukazuje na dobru kontrolu.



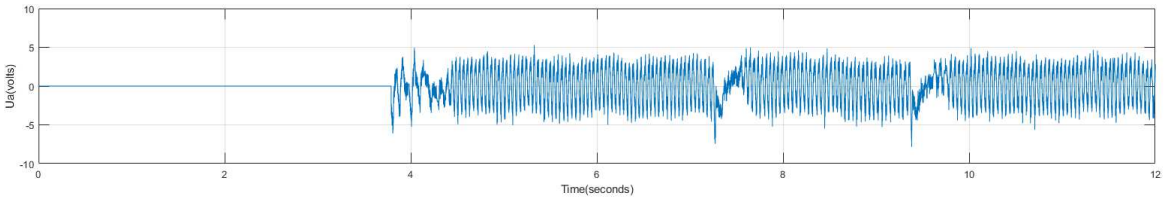
Slika 5.4 Graf ovisnosti kuta štapa o vremenu

Analizirajući graf koji prikazuje kut α u odnosu na vrijeme za stvarni postav, evo ključnih točaka analize:

Ključne točke analize:

1. **Početna faza**: U početnim sekundama kut α iznosi -180 stupnjeva, što ukazuje na početno stanje sustava tj. još uvijek nismo podigli štap iz ravnoteže. Zato u prvih nekoliko sekundi nema značajnih promjena.
2. **Nagla promjena**: Oko treće sekunde dolazi do brze promjene kuta α, s velikim porastom prema pozitivnim vrijednostima. Sustav pokazuje gotovo linearan porast sve dok ne dosegne približno 0 stupnjeva. Ova faza predstavlja podizanje štapa do neke točke te zatim korektivnu akciju sustava kako bi se kut α doveo u referentnu vrijednost, što ukazuje na snažnu kontrolu sustava.
3. **Stabilizacija**: Nakon što sustav postigne ravnotežnu vrijednost oko 0 stupnjeva, kut α ostaje relativno stabilan s vrlo malim oscilacijama. Oscilacije su prisutne, ali su

dovoljno male da možemo reći da je sustav u ovoj fazi relativno stabilan. Iako postoje male promjene, sustav uspješno zadržava kut blizu referentne vrijednosti.



Slika 5.5 Graf ovisnosti napona o vremenu

Analizirajući graf koji prikazuje napon Ua u odnosu na vrijeme za stvarni postav, možemo uočiti sljedeće:

Ključne točke analize:

1. **Početna faza**: U početnim sekundama napon Ua je stabilan i zadržava se na nultoj vrijednosti. Ovo sugerira da nema promjena u kontrolnim signalima ili da sustav čeka na odgovarajući signal prije početka korektivnih akcija..
2. **Nagla promjena**: Nakon treće sekunde dolazi do izrazitih oscilacija u naponskom signalu. Ove oscilacije počinju naglo, što ukazuje na pokušaj sustava da izvrši korektivnu akciju kako bi stabilizirao kut α. Oscilacije postaju sve učestalije i variraju između pozitivnih i negativnih vrijednosti, što ukazuje na konstantnu regulaciju i reakciju kontrolera na vanjske poremećaje.
3. **Stabilizacija**: Oscilacije se nastavljaju tijekom cijelog promatranog perioda od 4. do 12. sekunde. Iako oscilacije pokazuju varijacije u amplitudi, nije primjetan jasan trend stabilizacije ili smanjenja amplituda. To sugerira da sustav kontinuirano radi na održavanju stabilnosti.

Zaključci:

- **Dinamika sustava**: U sustavu podizanjem štapa nakon treće sekunde dolazi do naglih promjena u svim parametrima. Napon Ua počinje snažno oscilirati, što ukazuje na pokušaj sustava da odgovori na poremećaje i stabilizira kut. Ove oscilacije se nastavljaju sve do kraja simulacije, bez jasnog trenda smanjenja amplitude.
- **Stabilizacija**: Sustav reagira na poremećaje i pokušava stabilizirati kutove. Kut α postiže stabilizaciju s malim oscilacijama nakon korektivne akcije.
- **Performanse regulatora**: Regulator pokazuje djelomično zadovoljavajuće performanse. Sustav uspijeva postići stabilnost za kut α, ali promjene u naponu ukazuju na to da ima mjesta za poboljšanje regulatora.

5.4. Usporedba rezultata simulacije i stvarnog postava

U svakom eksperimentalnom sustavu, postoji razlika između simulacije i stvarnog postava zbog raznih faktora koji utječu na ponašanje sustava u realnim uvjetima. Jedna od najvećih razlika između simulacije i stvarnog postava je prisutnost šuma i smetnji u stvarnim uvjetima. U simulacijama, sustavi su idealizirani; nema vanjskih smetnji, trenja, vibracija, ili električnih smetnji koje mogu utjecati na ponašanje sustava. U stvarnim postavima, šumovi su neizbježni, bilo zbog elektroničkih komponenti ili okolišnih uvjeta. Ovi šumovi mogu uzrokovati veće i nepredvidive oscilacije u sustavu.

U stvarnim uvjetima, fizikalni fenomeni poput trenja i zračne otpornosti često nisu uključeni u simulacije. Simulacije obično pretpostavljaju savršene uvjete bez ovih dodatnih otpora, dok u stvarnom postavu ti faktori mogu značajno utjecati na dinamiku sustava. Na primjer, trenje između pokretnih dijelova u stvarnom postavu može uzrokovati sporiju stabilizaciju kutova θ i α, dok zračna otpornost može prigušiti ili pojačati oscilacije.

U simulacijama, algoritmi kontrolera su idealizirani i djeluju bez kašnjenja ili grešaka. Međutim, u stvarnim postavima, kontroleri mogu imati kašnjenja zbog fizičkih ograničenja hardvera, kao što su senzori ili aktuatori. Ova kašnjenja mogu utjecati na sposobnost sustava da pravovremeno reagira na promjene, što uzrokuje povećane oscilacije i sporiju stabilizaciju.

6. Zaključak

Ovaj rad je detaljno prikazao primjenu programskog sustava Matlab/Simulink u projektiranju i optimizaciji mehatroničkih sustava, s posebnim naglaskom na inverzno njihalo. Korištenje Matlab/Simulinka omogućilo je uspješno modeliranje, simulaciju i analizu ponašanja dinamičkog sustava inverznog njihala u kontrolnim aplikacijama. Kroz ovaj rad objašnjen je cjelokupan proces projektiranja, od matematičkog modela, preko simulacija, pa sve do implementacije regulatora u stvarnom sustavu.

Kroz simulacije je dobiven precizan i stabilan model koji prikazuje ponašanje sustava u idealnim uvjetima, dok je stvarni postav donio izazove zbog utjecaja različitih vanjskih faktora. U simulacijama su oscilacije bile prigušene i sustav je brzo stabilizirao parametre poput kuta θ i α. Međutim, u stvarnom postavu prisutni su elementi koji su uvelike otežali stabilizaciju, kao što su trenje, zračna otpornost, nelinearnosti u aktuatorima i senzorskim sustavima, kao i šumovi u električnim signalima.

Kroz rad je analizirana potreba za izračunavanjem optimalnog pojačanja K korištenjem LQR (Linear Quadratic Regulator) metode, koja je od ključne važnosti za stabilizaciju dinamičkog sustava. LQR metoda omogućila je optimizaciju povratne veze stanja, minimizirajući funkciju troškova koja balansira između preciznosti regulacije i minimalne potrošnje energije. Optimalno pojačanje K uspješno je primijenjeno u simulacijama, gdje je regulator pokazao izuzetno dobre rezultate, omogućujući stabilizaciju sustava s minimalnim oscilacijama i brzim odzivom.

U stvarnom postavu, međutim, unatoč korištenju optimalnog pojačanja K, sustav nije pokazao iste performanse kao u simulacijama. Razlog tomu su vanjski faktori kao što su trenje i šumovi, koji u simulacijama nisu prisutni. U stvarnim uvjetima, oscilacije su bile izraženije, a stabilizacija sporija. Iako je LQR omogućio solidnu kontrolu, dodatno podešavanje regulatora bilo bi potrebno kako bi se prilagodio stvarnim uvjetima i postigla stabilnost koju su pokazale simulacije.

Kroz ovaj rad pokazano je da Matlab/Simulink predstavlja neizostavan alat u projektiranju mehatroničkih sustava. Omogućuje brzu iteraciju kroz faze razvoja sustava, od matematičkog modeliranja do simulacije i optimizacije regulatora. Simulinkova sposobnost integriranja različitih fizikalnih domena, poput mehaničkih, električnih i dinamičkih, omogućila je preciznu analizu inverznog njihala i uspješnu validaciju upravljačkih algoritama. Uvođenje alata poput Simscape Multibody omogućilo je modeliranje složenih mehaničkih sustava, dok su alati poput Control System Toolboxa olakšali implementaciju i optimizaciju naprednih kontrolnih algoritama, poput LQR-a.

Unatoč napretku koji je postignut u ovom radu, preostaju izazovi u optimizaciji stvarnog sustava. Posebno se ističe potreba za daljnjom optimizacijom regulatora kako bi se poboljšala stabilnost u prisutnosti vanjskih smetnji, trenja i nelinearnosti. Budući rad mogao bi uključivati uvođenje naprednijih tehnika poput adaptivnih regulatora, robusne kontrole ili čak primjene algoritama umjetne inteligencije (AI) i strojnog učenja (ML) kako bi se povećala sposobnost regulatora da se prilagodi stvarnim uvjetima rada.

Literatura

[1] "Mehatronika". (n.d.). Preuzeto s https://tehnika.lzmk.hr/mehatronika/

[2] "What is MATLAB?". (n.d.). Preuzeto s https://www.mathworks.com/discovery/what-is-matlab.html

[3] "Simulink Documentation". (n.d.). Preuzeto s https://www.mathworks.com/help/simulink/

[4] "Cleve Moler Wins ICIAM Industry Prize 2023". (2022). Preuzeto s https://nhigham.com/2022/10/04/cleve-moler-wins-iciam-industry-prize-2023/

[5] "Mechatronics". (n.d.). U Wikipedia. Preuzeto s https://en.wikipedia.org/wiki/Mechatronics

[6] "Simulink". (n.d.). Preuzeto s https://in.mathworks.com/products/simulink.html

[7] "How Mechatronics Improve Drone Technology". (n.d.). Preuzeto s https://www.simplexitypd.com/blog/how-mechatronics-improve-drone-technology/

[8] "Simulink Block Diagram". (n.d.). Preuzeto s https://www.researchgate.net/figure/Simulink-block-diagram-The-reading-of-the-values-of-the-Matlabs-workspace-variables-is_fig11_228872583

[9] "Simscape Multibody". (n.d.). Preuzeto s https://www.mathworks.com/products/simscape-multibody.html

[10] "Simscape Electrical". (n.d.). Preuzeto s https://www.mathworks.com/products/simscape-electrical.html

[11] "Simscape Overview". (n.d.). Preuzeto s https://www.mathworks.com/videos/simscape-overview-61215.html

[12] "Control System Toolbox". (n.d.). Preuzeto s https://www.mathworks.com/products/control.html

[13] Bishop, R. H. (Ed.). (2017). *The Mechatronics Handbook*. 2nd Edition. CRC Press.

[14] Bolton, W. (2015). *Mechatronics: Electronic Control Systems in Mechanical and Electrical Engineering*. 6th Edition. Pearson.

[15] MathWorks. (2020). *MATLAB and Simulink for Control Design*. Available at: https://www.mathworks.com/solutions/control-systems.html

[16] Farid, A. M., & McFarlane, D. C. (2016). *Mechatronics: Principles and Applications*. Springer.

Sažetak

PRIMJENA PROGRAMSKOG SUSTAVA MATLAB/SIMULINK U PROJEKTIRANJU MEHATRONIČKIH SUSTAVA

Nikola Jamić

Ključne riječi:
Mehatronika, Matlab, Simulink, Modeliranje, Simulacija, upravljački algoritmi, Dinamički sustavi, Inverzno njihalo, Multibody, LQR

U ovom radu prikazana je primjena programskog sustava Matlab/Simulink u projektiranju i optimizaciji mehatroničkih sustava kroz primjer inverznog njihala. Cilj rada bio je izraditi model inverznog njihala, implementirati regulator pomoću LQR metode te analizirati razlike između simulacijskog modela i stvarnog laboratorijskog postava. Kroz simulacije, sustav je uspješno stabiliziran s minimalnim oscilacijama, dok je stvarni postav pokazao veće oscilacije zbog prisutnosti šuma, trenja i drugih vanjskih utjecaja. Optimalno pojačanje K izračunato je LQR metodom kako bi se postigao balans između stabilnosti i minimalne potrošnje energije. Iako su rezultati simulacija bili vrlo precizni, stvarni postav zahtijevao je dodatnu optimizaciju zbog stvarnih uvjeta rada. U radu su također obrađeni izazovi stvarnog postava te predložena poboljšanja u regulaciji sustava.

Summary

APPLICATION OF THE MATLAB/SIMULINK ENVIRONMENT FOR MECHATRONIC SYSTEM DESIGN

Nikola Jamić

Keywords:

Mechatronics, Matlab, Simulink, Modeling, Simulation, Control Algorithms, Dynamic Systems, Inverted Pendulum, Multibody, LQR

This paper presents the application of the Matlab/Simulink software in the design and optimization of mechatronic systems using the inverted pendulum as an example. The goal was to develop a model of the inverted pendulum, implement a controller using the LQR method, and analyze the differences between the simulation model and the real laboratory setup. In simulations, the system was successfully stabilized with minimal oscillations, while the real setup exhibited larger oscillations due to the presence of noise, friction, and other external factors. The optimal gain K was calculated using the LQR method to achieve a balance between stability and minimal energy consumption. Although the simulation results were very accurate, the real setup required further optimization due to real-world conditions. The paper also discusses the challenges of the real setup and suggests improvements in system regulation.

Privitak

Upute za korištenje programske podrške

Pokretanje simulacije

Za pokretanje simulacije potrebno je instalirati Matlab sa svim potrebnim alatima. Također su nam je potrebna datoteka Inverted_pendulum.slx tj. shema našeg modela. Nakon pokretanja u nju je potrebno upisati odgovarajući koeficijent K kojeg smo naveli gore u analizi. Taj K se može dobiti i pokretanjem priložene skripte parametri.m. Pokretanjem simulacije dobivamo grafički i vizualni prikaz postava inverznog njihala te možemo analizirati rezultate u različitim uvjetima.

Za spajanje sa pravim postavom potrebno je spojiti postav s računalom i u Matlab-u učitati "LAB5_ME13.slx" datoteku. Zatim pokrenuti u realnom vremenu i podiči štap sustava, zatim on kreće sa stabilizacijom.