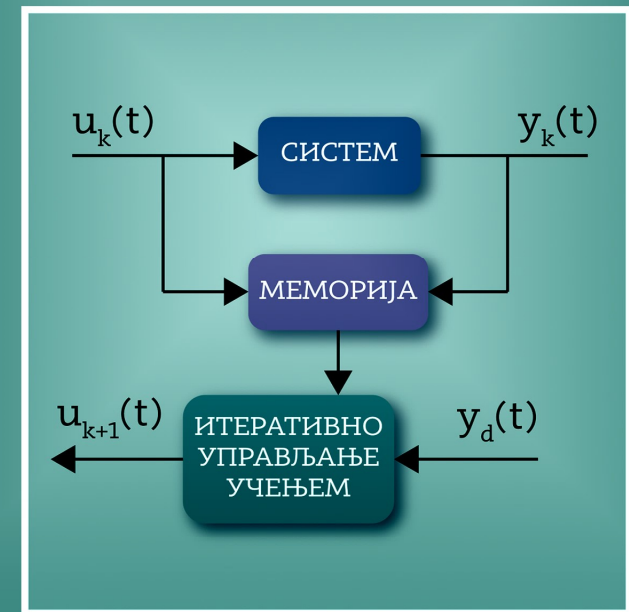


Михаило Лазаревић
Никола Живковић
Александар Дубоњац

ИТЕРАТИВНО УПРАВЉАЊЕ ПУТЕМ УЧЕЊА РОБОТСКИХ
СИСТЕМА: КЛАСИЧАН И МОДЕРАН ПРИСТУП

Михаило Лазаревић • Никола Живковић • Александар Дубоњац

Итеративно управљање путем учења роботских система: класичан и модеран приступ



ISBN-978-86-6060-081-5



Машински факултет
Универзитет у Београду, 2021

Михаило Лазаревић • Никола Живковић • Александар Дубоњац

**Итеративно управљање путем учења
роботских система: класичан и модеран
приступ**

**Машински факултет
Универзитет у Београду
2021**

Др Михаило П. Лазаревић, редовни професор
Машински факултет, Универзитет у Београду
Никола Живковић, мастер инж. маш.
Александар Дубоњац, мастер инж. маш.

**Итеративно управљање путем учења роботских система:
класичан и модеран приступ**

I издање

Монографија / Monograph

Рецензенти

Др Александар Обрадовић, ред. проф. Универзитет у Београду, Машински факултет
Др Томислав Шекара, ред. проф. Универзитет у Београду, Електротехнички факултет
Др Жељко Деспотовић, научни саветник Института Михајло Пупин,
Универзитета у Београду

Издавач

Универзитет у Београду
Машински факултет Београд, 11000 Београд, ул. Краљице Марије 16
tel. (011) 3370-760, fax. (011) 3370-364 www.mas.bg.ac.rs

За издавача

Декан, др Радивоје Митровић, ред. проф.
Уредник
Др Милан Лечић, ред. проф.
Председник комисије за издавачку делатност
Машинског факултета у Београду

Тираж: 200 примерака

Штампање I издања одобрила:
Комисија за издавачку делатност
Машинског факултета у Београду и
Декан Машинског факултета у Београду
Одлуком бр. 17/2021 од 23. 06. 2021.

Штампа: ПЛАНЕТА print

Стевана Томића бр.1, 11000 Београд
www.planeta-print.rs
Београд, 2021. године
ISBN-978-86-6060-105-8

Сва права задржавају аутори. Није дозвољено да без претходне писмене дозволе аутора било који део ове књиге буде снимљен, емитован или репродукован, укључујући, али не и ограничавајући се на фотокопирање, фотографију, магнетни или било који други вид записа.

Др Михаило Лазаревић, ред.проф.
Никола Живковић, мастер инж. маш.
Александар Дубоњац, мастер инж. маш.

**Итеративно управљање путем учења
роботских система: класичан и модеран
приступ**

Машински факултет
Универзитет у Београду
2021

Захвалност

*Издање ове научне монографије финансијски је помогло
Министарство за просвету, науку и технолошки развој Републике Србије,
на чему су Министарству аутори дубоко захвални.*

**Итеративно управљање путем учења роботских
система: класичан и модеран приступ**

САДРЖАЈ

1. Основе итеративног управљања учења регуларним системом целог реда

1.1 Основни појмови итеративног управљања учењем.....	1
1.1.1 Историјат и основне поставке итеративног управљања учењем.....	1
1.1.2 Различите врсте ILC управљања и њихови основни закони управљања.....	6
1.2 Основе линеарног итеративног управљања учењем.....	13
1.2.1 D–тип ILC-а за линеарни динамички систем.....	17
1.3 Увод у нелинеарно итеративно управљање учењем.....	19
1.4 PD-тип итеративног управљања учењем за линеарне континуалне системе са произвољним релативним степеном.....	22
1.4.1 PD-тип ILC-а за системе са релативним степеном два.....	22
1.4.2 PD-тип ILC-а за системе са произвољним релативним степеном γ	27
1.5 Један пример робусног итеративног управљања учењем.....	31
1.6 Један пример дискретног алгорита итеративног управљања учењем роботског система.....	37

2. ИЛЦ у ограниченом простору стања

2.1 Алгоритам ИЛЦ ограничене грешке (BEALC).....	49
2.1.1 BEALC-доказ конвергенције алгорита	52
2.1.2 Пример управљања робота применом BEALC	58
2.2 Алгоритам ИЛЦ ограниченог излаза COILC.....	64
2.2.1 COILC-доказ конвергенције алгорита	65
2.2.2 Пример управљања робота применом COILC.....	67
2.3 Резултати симулација – различите вредности параметара и поређење алгоритама.....	73
2.3.1 BEALC – други и трећи сет параметара.....	73
2.3.2 COILC – други и трећи сет параметара.....	76

3. Напредни итеративни алгоритам управљања учењем за рехабилитационе егзоскелетоне

3.1 Увод у рехабилитациону роботичку.....	80
3.1.1 Рехабилитациони работи горњих екстремитета.....	81
3.2 Биомеханички објекат управљања.....	82
3.2.1 Кинематичке и динамичке карактеристике биомеханичког система.....	83

3.2.2	Диференцијалне једначине кретања.....	85
3.3	Увод у управљање роботског система.....	86
3.3.1	Управљање у простору стања и у простору излаза.....	86
3.3.2	Синтеза трајекторије кретања у простору стања.....	88
3.3.3	Примена feedback линеаризације алгоритма управљања.....	91
3.3.4	Синтеза линеарног закона управљања-алгоритам управљања итеративним учењем.....	93
3.3.5	Симулациони резултати.....	97
4.	Напредни алгоритам робусног управљања ИЛЦ-а целог реда ПДД2/(П)ДД2 тип у отвореној-затвореној спрези	
4.1	Напредно моделирање роботског система.....	104
4.1.1	Нелинеарни математички модел роботског система.....	104
4.2	Пројектовање контролера применом feedback линеаризације.....	106
4.2.1	Робусни алгоритам AILC у отворено-затвореној спрези – PDD2/DD2 тип.....	108
4.3	Симулациони резултати.....	113
5.	Одабрани алгоритми ИУУ нецелог реда, типа D^α , PD^α	
5.1	Оптимално ИУУ нецелог реда типа D^α , PD^α , $\alpha \in [0,2]$ роботског система.....	126
5.1.1	Структура робота и једначине кретања.....	127
5.1.2	Улаз-стање линеаризујући контролер.....	128
5.1.3	Анализа конвергенције предложеног контролера нецелог реда.....	131
5.1.4	Оптимизација FOILC користећи ВВО алгоритам.....	132
5.1.5	Симулациони резултати.. ИУУ нецелог реда.....	135
5.1.6	Симулациони резултати -ИУУ нецелог реда са ВВО алгоритмом.....	142
5.2	ИУУ за нелинеарне системе нецелог реда.....	145
5.2.1	П-тип ИЛС алгоритам.....	146
5.2.2	Нумеричка симулација.....	152
5.3	ИУУ за нелинеарне системе нецелобројног реда са кашњењем.....	154
5.4	PD^α тип ИЛС са променљивим појачањем у повратној грани.....	160
5.4.1	Нумеричка симулација.....	168

6. Примена ИУУ нецелог реда сингуларним системом нецелог реда који укључују и временско кашњење

6.1 Итеративно управљање путем учења нецелог реда за линеарне сингуларне системе нецелог реда.....	175
6.2 Итеративно управљање путем учења ПД типа нецелог реда за сингуларни систем нецелог реда са кашњењем.....	176

7. ИУУ типа PD^2D^α нецелог реда у отвореној петљи линеаризованим динамичким системом

7.1 Резултати конвергенције -закон учења PD^2D^α у отвореној петљи.....	188
--	-----

ДОДАЦИ

Додатак А Преглед најважнијих норми.....	197
Додатак Б Основе фракционог рачуна (<i>рачуна нецелог реда</i>) - дефиниције најзначајнијих фракционих оператора	199
Додатак Ц Основни појмови сингуларних система.....	217
Додатак Д Математички модел роботског система.....	224
Додатак Е Математички модел NeuroArm роботског манипулатора.....	239
Додатак Ф <i>Feedback</i> линеаризујући системи управљања.....	254
ЛИТЕРАТУРА	271

ПРЕДГОВОР

Примена и присуство фракционог рачуна у свим гранама науке и технике више је него евидентно јер омогућава да се уочени системи квалитетније, боље моделирају, односно развију и примене квалитетнији системи управљања. У том смислу бројни научни радови и обимна публицистичка делатност у пуној мери су исказали интерес који је за њих био показан. У математичком смислу, за разлику од „класичног“ диференцијалног и интегралног рачуна, овде степен може бити реалан број чак и комплексан број, тако да су одговарајући оператори диференцирања и интегралнења нелокални оператори и дати системи се сад могу описати са диференцијалним једначинама али фракционог реда тако да систем има више степени слободе. У специјалном случају када је степен целобројан онда се претходни оператори свODE на „класичне“ операторе диференцирања и интегралнења.

Један од циљева данас у савременој теорији управљања свакако јесте примена фракционог рачуна и то регуларним и сингуларним динамичким системима целог реда а посебно фракционог реда. Осим тога, као што је познато, континуални сингуларни системи целог реда у математичком смислу, представљају динамичке системе описане комбинацијом алгебарских и диференцијалних једначина, што не дозвољава њихово представљање у класичном облику векторске диференцијалне једначине стања, а самим тим и онемогућава решавање тог система једначина уобичајеним методама које се користе за решавање “нормалних” система. Посебан изазов овде представљају сингуларни системи фракционог реда са и без временског кашњења.

Посебна пажња је овде посвећена на напредној примени интелигентног управљања путем учења (ИУУ) које представља моћан интелигентни концепт управљања који на итеративни начин побољшава понашање процеса који имају особину понављања у својој природи. Прву примену оваквог типа учења је остварена на решавању задатка управљања датим роботским системом. Главни акценат је овде дат на примени ИУУ целобројног реда који укључују и изводе грешке вишег реда као и алгоритме ИУУ-а фракционог реда за управљање динамичких роботских система тако и за сложеније системе фракционог реда који укључују нелинеарности и временско кашњење.

Посебан квалитет у првом делу ове монографије представљају добијени резултати који се односе на примену напредног итеративног алгоритма управљања учењем за рехабилитационе егзоскелетоне као и примену алгоритама ИУУ целобројног реда типа ПДД2 и то за постојеће роботске системе са тежњом ка практичној реализацији чиме ова монографија добија на свом значају.

У другом делу монографије, презентована је примена савремене теорије рачуна нецелог реда (*fractional calculus*-а) у задацима управљања кретањем. Прво су презентовани одабрани алгоритми ИУУ нецелог реда. Са не мањим значајем по први

пут презентована је примена робусног алгоритма ИЛЦ фракционог реда сингуларним фракционим системима са кашњењем као један нови алгоритам ИУУ фракционог реда пројектован за разматрани линеаризовани динамички роботски систем чиме се отвара могућност за даљу практичну реализацију истог.

Изложена материја подржава актуелне трендове у овај области и предствља селективан текст који садржи и нека теоретска знања преузета из савремених уџбеника из ове области а све са циљем да се заинтересованом читаоцу омогући лакши и непосреднији приступ овој сложеној проблематици.

У овој монографији представљени су досадашњи најзначајнији резултати из области итеративног управљања путем учења првопотписаног аутора а који су проистекли из сарадње са докторантима и колегама који се баве овом проблематиком.

Др Александру Обрадовићу, редовном професору Машинског факултета Универзитета у Београду, др Томиславу Шекари, редовном професору Електротехничког факултета Универзитета у Београду и Жељку Деспотовићу, научном саветнику Института Михајло Пупин захваљујемо се на корисним сугестијама и труду око рецензије ове монографије.

За рад око техничке обраде монографије захвални смо колеги Илији Лазаревићу, мастер инж. маш.

А у т о р и

Београд, јул, 2021. год.

1. Основе итеративног управљања учења регуларним системом целог реда

1.1. Основни појмови итеративног управљања учењем

1.1.1 Историјат и основне поставке итеративног управљања учењем

Јапански научник Учијама (*Uchiyama*) је 1978. По први пут представио идеју о итеративном управљању учењем, тј. почетну експлицитну формулацију ИЛЦ-а. али је рад био написан на јапанском, због чега његов утицај није био толико видљив у научној литератури. Касније, 1984. године (*Аримото et al, 1984*) представили су ИЛЦ методу на енглеском језику где је предложено ИЛЦ управљање за прецизно праћење роботских трајекторија. Оно се односило на методу управљања која непрекидно понавља покушаје управљања са истом жељеном трајекторијом и коригује закон управљања како би остварио задовољавајуће управљање. Њихова мотивација у примени ИЛЦ-а лежи у чињеници да индустријски роботски манипулатор у обављању одговарајућих манипулационих задатака понавља исти задатак из више покушаја. Да би се превазишао овај проблем, Аримото, је предложио да информације из претходних покушаја и перформанси самог задатка буду искоришћене да побољшају само управљање током текућег покушаја. Он је назвао ову методу “*betterment process*”.

Важно је поменути да у 1984. неколико тимова је пријавило да поновљивост система може бити искоришћено за побољшање перформанси. То су били:

- *S. Arimoto, S. Kawamura and F. Miyazaki, 1984*
- *J. J. Craig, 1984*
- *R. H. Middleton, G. C. Goodwin and R. W. Longman, 1989*
- *E. G. Harokopos, 1986.*

ILC је у 20. веку добио 80 нових теорија управљања, захваљујући својим значајним предностима, и током последњих година је у великом развоју. *ILC* је погодан за управљање објеката који врше поновљива кретања, и који се извршава у ограниченом временском интервалу. Покушава да коригује већ управљан систем, коригује управљачки систем користећи сигнал грешке, генерише нови управљачки сигнал, и побољшава перформансе праћења система.

Сам термин “*учење*”, обично значи или стицање или трансфер знања. Међутим ту се дефинисање тог појма не завршава. У нашем језику, учење се може дефинисати и као:

Процес који мења понашање јединке на основу претходног искуства.

Такође, учење може да представља и *процес стицања новог или модификовања постојећег знања, понашања, вештина, вредности, итд.*, као и то да учење има много подврста, као и различите методе које оно обухвата, у зависности од циљева и потреба

као и могућности. Како је тема рада везана за итеративно управљање учењем (*ILC – Iterative Learning Control*), превасходно ће фокус бити на томе.

Реч “*iterative*” – у енглеском језику означава радњу која захтева да се динамички процес понавља или: *динамички систем је детерминистички и задатак управљања праћењем се понавља на коначном временском интервалу*, (*Jian-Xin Xu, Ying Tin, 2003*).

Сама потреба за ILC-ом, јавља се из потребе за већим (дубљим) разумевањем тј. препознавањем да се знање може стећи (научити) из искуства. Зато ILC често називамо - *искусствено учење*. Ово новостечено *искуство* служи као нов извор знања за дати модел објекта динамичког система. Тиме ново знање стечено искуственим учењем, ствара нове могућности у циљу побољшања управљања.

Итеративно управљање путем учења (ИЛЦ- *iterative learning control*) је једно од атрактивних области у теорији управљања и оно представља моћна интелигентна методологија управљања који на итеративни начин побољшава понашање процеса који имају особину понављања у својој природи, *Moore (1993), Bien and Xu(1998), Ahn et al.(2007)*. Наиме, ИЛЦ је релативно нова али већ афирмисана област изучавања у теорији управљања. ИЛЦ, које се може сврстати у методологију интелигентног управљања, је метод за побољшање непостојаних перформанси поновљивих система у једном одређеном временском интервалу. Системи за које се обично сматра да раде под ИЛЦ су поновљиви динамички системи, као што је роботски манипулатор у производном окружењу или хемијски реактор. Појам ИЛЦ се такође може проширити на динамичке системе са периодичним поремећајима или периодичним кретањем, где периодичност може да зависи од времена, стања или трајекторије. Или, речено уопштеније, основна идеја ИЛЦ се може схватити као вишепролазни процес.

Иако теорија управљања пружа бројне алате за побољшање одзива динамичког система, није могуће увек постићи потребне жељене перформансе услед присуства немоделиране динамике или параметарских несигурности које се испољавају за време тренутног рада система или због недостатка одговарајућих техника пројектовања. Зато није лако постићи савршено праћење коришћењем класичних теорија управљања и познато је да обични класични алгоритми управљања не могу искористити особину поновљивости извођења истог задатка. Са друге стране ИЛЦ припада класи напредних алгоритама управљања који се може користити за превазилажење недостатака класичног конструисања контролера, нарочито да би се добио жељени пролазни одзив за специјални случај када је систем о коме је реч поновљив. За такве системе, ИЛЦ се често може користити да би се постигло савшено праћење, чак и када је модел несигуран или непознат и немамо информација о структури и нелинеарности система.

У литератури постоје различите дефиниције ИЛЦ, види (Ahn, et al. 2007) али оно што им је заједничко јесте идеја о „понављању“. Другим речима, за разлику од класичних алгоритама управљања, ИЛЦ представља једноставно и ефикасно управљање које може прогресивно смањивати грешке праћења и истовремено побољшати перформансе система из итерације у итерацију, Casalino and Bartolini (1984), Bien and Huh (1989). ИЛЦ приступ представља у основи више или мање имитацију процеса учења који карактерише свако интелегентно биће. Интелигентна бића имају тенденцију да уче изводећи покушај (тј одабирају управљање) и посматрају шта је био крајњи резултат таквог избора управљања.

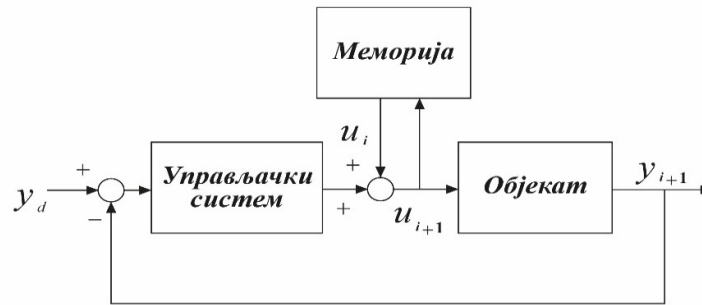
Другим речима, основна разлика између управљања учењем од многих других постојећих управљања је што је учење у суштини *off-line* процес где постоји коначно време између два покушаја. При томе управљање учењем представља комбинацију управљања у директној („*feedforward*“) спрези и управљање у повратној спрези при чему доминантну улогу представља управљање у директној спрези и као такво оно има за циљ да оствари *предикцију* будућег управљања. Предикција је овде последица упамћеног знања стеченог учењем. Уочено је да је управљање у директној спрези заступљено у процесу управљања живог организма у природи. Такав механизам омогућава предикцију наредних покрета организма а која је заснована коришћењем претходног знања и искуства. Осим тога, може се рећи да је ИЛЦ је рекурзиван метод управљања који захтева мањи број израчунавања као и да захтева мање унапред познатих података о објекту управљања него многе друге врсте управљања.

Наиме, уочено је да код извођења манипулационих задатака који се састоје у понављању једне исте трајекторије, односно код цикличних покрета, применом управљања на основу итеративног учења, могуће побољшати кинематичке и динамичке карактеристике датог система. Најчешће, то су брзина праћења трајекторије и прецизност извођења задате трајекторије где се побољшање истих реализује коришћењем задатог закона управљања на основу итеративног учења. Основна одлика закона управљања на основу итеративног учења је да се у текућој итерацији-покушају, вектор управљања генерише на основу вектора управљања из претходне итерације-покушаја и информације о грешци праћења жељене излазне трајекторије из претходне итерације. На тај начин, учи се на основу претходног покушаја и искуства.

Праћење подразумева да жељена трајекторија стања буде праћена од њеног самог почетка па до краја кретања система. С друге стране, понављајуће управљачко окружење, подразумева сличну трајекторију стања и сличне почетне услове за сва испитивања таквог управљања. Многе постојеће методе управљања нису у стању да испуне овакав задатак јер њихов циљ је асимптотска стабилност (конвергенција) али оно најважније – те методе не могу да “уче” из претходних покушаја управљања, без обзира да ли је оно успешно или не. То значи да, без учења, систем није способан да

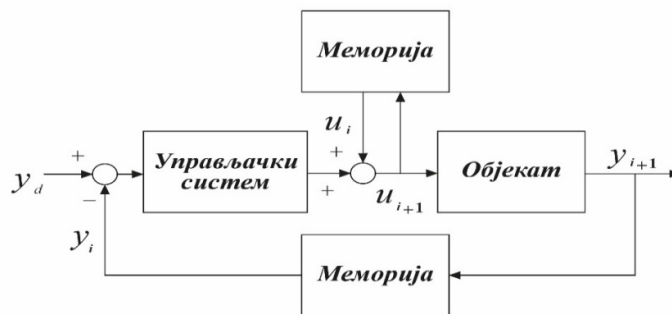
ради боље, нити да побољша своје перформансе, упркос задатку који се понавља (цикличан је). Даје се ILC чија је основна идеја, види слику 1:

-да се искористи информација управљачког система претходног покушаја (итерације) да би се побољшале управљачке перформансе тренутне итерације. Ово је омогућено кроз тзв. меморијско учење.



Сл. 1.1 Основна шема итеративног управљања путем учења

На слици се види да поред стандардне *feedback* петље, има и меморијска компонента која служи да сними управљачки сигнал претходног покушаја, $u_i(t)$, који је део садашњег управљачког сигнала, $u_{i+1}(t)$.



Слика 1.2. Типичан изглед ILC-а

Нека је за уочени систем позната и задата жељена трајекторија $y_d(t)$ у простору излаза. Такође, грешка праћења у i -тој итерацији се може дефинисати са:

$$e_i(t) = y_d(t) - y_i(t) \quad (1.1)$$

где је са $y_i(t)$ означен вектор излаза уоченог система и $t \in [0, T]$ где је време извођења покрета T задато. Задатак је да се применом концепта итеративног учења за познату жељену излазну трајекторију $y_d(t)$, одреди вектор управљања $u_i(t)$ такав да када број итерација $i \rightarrow \infty$, $\lim_{i \rightarrow \infty} u_i(t) = u_z(t)$, $\forall t \in [0, T]$, вектор излаза система $y_i(t)$ прати жељену излазну трајекторију са задатим степеном тачности, тј. $\lim_{i \rightarrow \infty} \|e_i(t)\| = \varepsilon_e$. Као што је претходно истакнуто, итеративно управљање учењем је први разматрао и увео Аримото, Arimoto *et al.*, (1984), где је разматран једноставан алгоритам учења Д типа у временском домену који захтева познавање првог извода сигнала грешке $\dot{e}_i(t)$. Касније, други аутори су предложили комплексније алгоритме учења (на пример, види Bien, Huh (1989)). Основна стратегија ИЛЦ се може приказати у следећој форми:

$$u_{i+1}(t) = f(u_i(t), e_i(t)), \quad e_i(t) = y_d(t) - y_i(t) \quad (1.2)$$

где $f(\dots)$ дефинише алгоритам за учење и остаје да се специфира, $y_i(t)$ је излаз у i -тој операцији која је настала као резултат дејства управљања $u_i(t)$ и $y_d(t)$ које представља жељени излаз. Ново управљање $u_{i+1}(t)$ треба реализовати тако да систем буде ближе жељеном циљу у наредном извршењу. Као што је то раније истакнуто, Аримото је у циљу управљања датим роботским манипулатором предложио следећи тип ИЛЦ управљања тзв. Д-тип.

$$u_{i+1}(t) = u_i(t) + \Pi \frac{de_i(t)}{dt}, \quad (1.3)$$

где је са i - означен број итерације (покушаја), $e(t)$ представља грешку праћења (1.1); Π је тзв. појачање учења и одређује се на основу претходног знања о систему. Овде $t \in [0, T]$, где T представља коначно време извођења датог задатка које је најчешће познато и коначно. Основна идеја ИЛЦ је приказана на сл. 1.1 где су $u_k(t)$ односно $y_k(t)$ улаз и излаз система код k -те итерације, $u_{k+1}(t)$ је улаз система код $k+1$ -тог покушаја, а $y_d(t)$ је дата жељена трајекторија. Циљ ИЛЦ је да је $\lim_{k \rightarrow \infty} y_k(t) = y_d(t)$ за свако $t \in [0, T]$, где је T фиксна константа. Предности ИЛЦ алгоритма су приказане у његовој примени на нелинеарне системе и системе са информацијама несигурне и непознате структуре, итд. (Arimoto *et al.*, 1984).

1.1.2 Различите врсте ILC управљања и њихови основни закони управљања

Постоје различите ILC шеме које ће бити разматране, укључујући *online* и *offline* варијанте са променљивим појачањима. *Online ILC*, *Offline ILC*, и *online-offline ILC* управљачки модели укључују и П, Д, и ПИД управљачке законе (Zhang et al, 2021). Када се за управљање користи ILC, постоје одређене ставке на које треба обратити пажњу, као што су прецизност, брзина конвергенције, компензација поремећаја и неизвесности.

Offline P-tun ILC

Приказан је закон управљања за П-тип ILC-а. Следећи израз (1.4) демонстрира једноставан закон учења:

$$\bar{u}_{k+1}(t) = \bar{u}_k(t) + K_{pof} \hat{e}_k(t) \quad (1.4)$$

где $\bar{u}_{k+1}(t)$ је ажурирани улаз (управљање) и $\hat{e}_k(t)$ је грешка. Стандардни услов за конвергенцију ILC-а П-типа описан је следећом једначином (1.5)

$$\bar{\rho}_{pof} = \|I - K_{pof}CB\| < 1 \quad (1.5)$$

$\bar{\rho}_{pof}$ је конвергенција а K_{pof} је појачање, и ако је $\bar{\rho}_{pof}$ мање од 1 грешка праћења конвергира, у супротном она дивергира. Овај услов мора бити испуњен за успешну конвергенцију грешке праћења.

Offline ILC закон учења са ПИД дејством

Слично претходно размотреним offline П ILC-ом, остали типови дејстава, Д, ПД и ПИД могу бити коришћени са одговарајућим појачањима за сваки од њих. Једначина (1.6) представља основни offline закон ажурирања:

$$\bar{u}_{k+1}(t) = \bar{u}_k(t) + K_{pof} e_k(t) + K_{iof} \int \hat{e}_k(t) dt + K_{dof} \bar{e}_k(t) \quad (1.6)$$

где $\bar{u}_{k+1}(t)$, $e_k(t)$, K_{dof} су ажурирано управљање, грешка и параметар учења, респективно.

Диференцијално појачање је исто за све претходно поменуте законе, а закон конвергенције је описан следећом једначином (6):

$$\bar{\rho}_{dof} = \|I - K_{dof}CB\| < 1 \quad (1.7)$$

Online закон управљања учењем

Дати израз (1.8) описује закон управљања за online П –тип ILC:

$$\bar{u}_{k+1}(t) = \bar{u}_k(t) + K_{pon} \hat{e}_{k+1}(t) \quad (1.8)$$

Претходни израз (1.9) представља *online* П-тип закон где је $\hat{e}_{k+1}(t)$ грешка у наредној итерацији. Такође, услов конвергенције је дат са (1.9):

$$\bar{\rho}_{pon} = \|(I + K_{pon}CB)^{-1}\| < 1 \quad (1.9)$$

Online ILC закони учења за све типове дејстава

Online ILC свих дејстава укључујући П, Д, ПД, ПИД дејства даје боље резултате праћења у поређењу са offline ILC-ом.

$$\bar{u}_{k+1}(t) = \bar{u}_k(t) + K_{pon}\hat{e}_{k+1}(t) + K_{ion} \int \hat{e}_{k+1}(t)dt + K_{don}\bar{e}_{k+1}(t) \quad (1.10)$$

Слично горе поменутиим законима према (1.10) услов конвергенције је дат са:

$$\bar{\rho}_{don} = \|(I + K_{don}CB)^{-1}\| < 1 \quad (1.11)$$

Као што се може видети из претходних израза (1.9) и (1.10), параметри учења (појачања) су независна. То значи да се одвојено могу подешавати одговарајући параметри према систему.

Такође, можемо их поделити и на временски континуалне и временски дискретне алгоритме ИЛЦ. Конвенционална шема ИЛЦ-а реализује управљање у тзв. *отвореној петљи*, (*off-line* ILC) која допуњује текући вектор управљања само са информацијом из претходних итерација и следствено не може побољшати перформансу система само у једном покушају односно итерацији. Ако пођемо од класичног Аримото типа ИЛЦ алгоритма може се развити неколико општијих израза. На пример, алгоритам ИЛЦ-а ПИД тип се може дати као (*Moore, 1993*):

$$u_{k+1} = u_k + \Phi e_k + \Gamma \dot{e}_k + \Psi \int e_k dt \quad (1.12)$$

где су Φ , Γ и Ψ матрице појачања учења. ИЛЦ вишег реда - што значи да се информација из више од једног претходног покушаја користи код ИЛЦ алгоритма – ажурираног правила попут ПИД (*Chen, Wen, 1999*) може формулисати као

$$u_{k+1} = \sum_{k=1}^N (I - \Lambda) P_k u_k + \Lambda u_o + \sum_{k=1}^N \left(\Phi_k e_{i-k+1} + \Gamma_k \dot{e}_{i-k+1} + \Psi_k \int e_{i-k+1} dt \right) \quad (1.13)$$

Ако је $\sum_{k=1}^N P_k = I$, онда правилним избором матрица појачања учења можемо да обезбедимо да e_k асимптотски тежи нули. Слично томе, временски променљива П

типа (што значи без утицаја извода и интеграла) верзија новог правила ИЛЦ, дата у једн. (1.14), се може написати као

$$u_{k+1}(t) = u_k(t) + \Gamma_k(t)(y_d(t) - y_k(t)) \quad (1.14)$$

где је $\Gamma_k(t)$ пропорционална матрица појачања учења која је сада временски променљива. Код овог ИЛЦ алгоритма првог реда, правилним избором матрице појачања учења $\Gamma_k(t)$, ИЛЦ процес ће тежити нултој грешци стационарног стања за системе релативног нултог степена. Слични резултати се могу развити за системе релативног степена један или вишег. Код овог једноставног ИЛЦ алгоритма, основна карактеристика ИЛЦ јесте да то управљање користи информацију из најскоријег прошлог покушаја за тренутно ажурирање. Отуда је такође природно извести ажурирана правила за временски променљив, вишег реда ИЛЦ као што је

$$u_{k+1}(t) = u_k(t) + \sum_{i=k}^{i=k-1} \Gamma_i(t)(y_d(t) - y_i(t)) \quad (1.15)$$

или

$$u_{k+1}(t) = \sum_{i=k}^{i=k-1} A_i(t)u_i(t) + \sum_{i=k}^{i=k-1} \Gamma_i(t)(y_d(t) - y_i(t)) \quad (1.16)$$

која не користе само информације о најскоријим грешкама улазног/нестабилног управљања већ све информације о претходним грешкама улазног/нестабилног управљања. Ови алгоритми истичу став да је ИЛЦ „закон управљања који користи све доступне информације из прошлости за побољшање перформансе периодичног система.“ Такође, могу се користити и шеме ИЛЦ-а које су вишег реда и извода и које могу остварити боље перформансе у решавању задатка праћења жељене трајекторије, *Lazarević (2003b)*.

$$u_{i+1}(t) = u_i(t) + \Gamma CB \frac{du_i(t)}{dt} + \Gamma \left(\frac{d^2 e_i(t)}{dt^2} + Q \frac{de_i(t)}{dt} + R e_i(t) \right) \quad (1.17)$$

Наиме, најновији трендови развоја и примене роботике указују све већу директну кооперацију између човека и робота у блиској будућности, *Potkonjak, Lazarević et al.,(2001)*. Као такви они треба да имају карактеристике сличне људским што се тиче способности и понашања, односно управљање таквим антропоморфним роботима засниваће се на примени одговарајућих биолошких аналога, (*Potkonjak, Lazarević et al.,(2001), Lazarević,(2001)*). Управљање роботским системом може се реализовати и на основу итеративног учења које заузима значајно место у биолошким и техничким системима, *Lazarević (2002)* где се примена ИЛЦ алгоритма управљања може применити на дати роботски систем без обзира да ли исти поседује кинематичку редундансу или не, *Lazarević (2003b)*. Ако у извршавању одређених задатака роботски

систем у доста дугом временском периоду понавља задате операције, онда постоји основа за увођење и синтезу управљања на основу процеса учења.

Већина ПЛС варијанти које су развијене до данас, приказане су на слици 1.3. Комбиновање ПЛС-а са другим управљачким методама и искоришћење предности ПЛС-а је одувек био правац којим су многи истраживачи кренули. Код оптималног управљања, дизајн контролера треба бити базиран на прецизном математичком моделу система. Ако се модел система промени, оптимални контролер више неће бити оптималан. Комбиновањем оптималног и итеративног управљања учењем брзина конвергенције ПЛС-а може бити увећана, тачније, усвајањем одређење функције циља и методе оптимизације у циљу синтезе оптималног закона управљања (Xu, Yang, 2013). У сфери адаптивног управљања, учење се односи на *online* корекције контролера. Иако је адаптивно управљање прилагодљиво променама средине и неосетљиво на грешке у моделовању, највећа слабост се огледа у захтеву за великом количином ресурса за *online* идентификацију параметара. Израчунавање у ПЛС-у је *offline* процес учења, током којег смањује грешку имитирањем људског учења како би стекао искуство из поновног извршења истог задатка. Према томе, постоји велики практични значај у комбиновању ПЛС-а и адаптивног управљања, (Tafti, Sangwongwanich, 2019) (Ma, Bi, 2019).

Робусно управљање може гарантовати стабилност система и одржати одређени индекс перформанси унутар одређених граница фактора неизвесности, самим тим решавајући проблем неизвесности модела објекта управљања и неизвесности спољних поремећаја, (Chen et al., 1997). Адаптивно управљање и робусно управљање деле неизвесности система на структурне и неструктурне неизвесности. Већина ПЛС метода не узима у обзир директно неструктурне неизвесности, али ПЛС не зависи од прецизног математичког модела система. На коначном временском интервалу, нелинеарни високо спрегнути динамички систем са великом неизвесношћу може са лакоћом бити управљан и може прецизно пратити задату путању. Овакав начин управљања се успешно користи у области система са управљањем кретања.



Слика 1.3 Врсте ILC управљања

Остале врсте ILC-а

Исто тако, класични ILC може бити коришћен само за посебан проблем праћења трајекторије. Резултат учења је процес управљања вођен из сачуване меморије. Чим је потребно променити излазну трајекторију, нова трајекторија мора бити поново научена. Дакле, то представља вид "учења напамет". Овакво знање не може бити примењено у новим ситуацијама, што ограничава његов обим примене. Комбинација итеративног управљања учењем и интелигентног управљања као што су експертски системи и неуронске мреже могу надоместити овај недостатак ILC-а, као и побољшати интелигенцију система извлачењем карактеристичних информација из резултата учења.

Најновија истраживања везана за ILC-а

- Умрежени Multi-Agent системи
- Итеративно колаборативно управљање (са човеком)
- Human in the loop итеративно управљање
- Машинско учење у ILC-у



Слика 1.4 Будуће области примене ILC-а

Умрежени Multi-Agent системи (Network Multi-Agent systems - NMAS)

Умрежени multi-agent системи (NMAS) могу бити употребљени када је потребна убрзана конвергенција и компензација кашњења у комуникацији. Системи од више робота и сензорски комплексни системи са применом усмеравају све више пажње ка колаборативним multi-agent системима у последњих пар година (Kebir, et al. 2018). У умреженим multi-agent системима сагласност представља заједничку тачку поклапања физичке величине уз помоћ података обезбеђених од стране одговарајућег протокола сагласности од локалних суседа, у којима је динамика тачака узета у обзир према топологији мреже (Su et al. ,2017).

Недавно појавио се проблем праћења multi-agent система и привукао је многе истраживаче посебно фокусиране на сагласност у праћењу заснованом на дистрибуцији на ограниченом временском интервалу за нелинеарне multi-agent системе. Истраживачи су открили да дефинитна област може бити надгледана кроз побољшану управљачку методологију заснованом на прикупљеним подацима. Ово је

најбољи пример система са кашњењем у комуникацији за умрежене нелинеарне multi-agent система са проблемом праћења заснованим на сагласности (Zhang et al, 2012).

Итеративно колаборативно учење (са човеком)

Недавно сарадња човека и робота (*Human-robot collaboration - HRC*) у медицинским и у инжењерским апликацијама постала је популарна област за истраживаче. Са напретком процеса производње у индустрији сложеност фабрика које врше задатке који укључују сарадњу човека и машине се повећава, неки од њих су аутоматске машине, интерфејс човека и робота, итд, (Dani, Mehta, 2014). На пример, у раду (Ravichandar, et al. 2019) аутори су предложили и увели ИЛЦ контролер базиран на естимацији за покрете људске руке и развили контролер колаборативног робота чије појачање представља укупно појачање покрета руке и робота.

Human in the loop итеративно управљање

Такође, human in the loop управљање је једна од будућих примена у пољу роботике, као израда интерфејса човека и машине унутар окружења виртуелне реалности (VR) о којем је дискутовано. Више различитих *Human in the loop* метода итеративног управљања учењем је недавно представљено али у овом истраживању, полу-аутономна метода је усвојена за извршење задатка интеракције. Задатак је већински изведен и испитан у окружењу виртуелне реалности (VR). Међутим, управљивост смањена полу-аутономним управљањем представља ману корисничког искуства. Свакако, треба ценити важно актуелно истраживање у области *human in the loop* итеративног управљања и не треба игнорисати визуелизацију у окружењу виртуелне реалности (Fröhner et al. 2019).

Машинско учење у ИЛС-у

Машинско учење унутар ИЛС-а, као код уграђеног итеративног контурног учења EILCC (*embedded iterative learning contour control*, - уграђено итеративно контурно управљање учењем) и конвенционалног итеративног контурног учећег контролера (CILCC) може бити коришћено када се од система захтевају тачност и прецизност. На пример, велики број укрштених (упарених) метода су побољшане ради управљања грешком одређених оса услед брзих повратних грана као што је брзина упарене адаптивне повратне спреге и упареног ИЛС-а. Аксијално кретање је кључан фактор за перформансе праћења трајекторије машина алатки које није само повезано са аксијалном грешком код обратка, већ и за контурне грешке. Грешке праћења и контурне грешке су измерене и умањене применом EILCC методе. Даље, експериментални резултати EILCC упоређени са конвенционалним итеративним контурним учећим контролером показали су да су резултати знатно побољшани (Paul, et al. 2019).

2.2. Алгоритам ограниченог излаза (COILC)

Алгоритам ограниченог излаза, као што само име каже, директно ограничава максималну вредност излаза на вредности Q_i^{min} и Q_i^{max} . За разлику од ВЕА који присиљава трајекторију да се током читавог процеса праћења налази унутар хипер цилиндра полупречника ε , који је дефинисан као најмање безбедно растојање од границе генерализаних координата, СО ограничава само максималну вредност излаза допуштајући одступања у деловима трајекторије где је то безбедно што би иначе код ВЕА било непотребно ограничено.

Такође, у комбинацији са COILC алгоритмом за формирање управљања у главној грани (*feedforward*) $u_k(t)$, користиће се предложено управљање (2.9) у повратној грани (*feedback*).

Кроз кораке COILC се може представити (Yovchev, Delchev, Krastev 2020.):

1. Поставити почетни број итерација $k = 0$ и започети итеративни поступак.
2. Из почетног положаја $q_k(0) = q_d(0)$ робот прати жељену трајекторију под дејством управљања $u(q, t) = u_k(t) + u_{fb}(t)$ током којег је $Q_i^{min} < q_i^k < Q_i^{max}$, $i = 1, 2, \dots, n$ и $t < T$, све до тренутка T_k : $0 < T_k < T$ у ком је $q_i^k = Q_i^{min}$ или $q_i^k = Q_i^{max}$, или док не достигне крајњи положај $q_k(T)$, $t = T$ када се процес праћења завршава.
3. Након завршене итерације, управљање у главној грани се ажурира за наредну итерацију према следећем закону:

$$u_{k+1}(t) = u_k(t) + \begin{cases} L(q_k(t))[\ddot{q}_d(t) - \ddot{q}_k(t) + L_v(\dot{q}_d(t) - \dot{q}_k(t)) \\ \quad + L_p(q_d(t) - q_k(t))], t \in [0, T_k]; \\ 0, t \in (T_k, T] \end{cases} \quad (2.41)$$

4. Ако је грешка праћења мања или једнака прихватљивој тачности и тренутак T_k је једнак T онда је процес учења успешно завршен и пронађено је жељено управљање које је једнако u_k последње итерације. У противном, увећати број итерација $k = k + 1$ и поново се вратити кораку 2.

Овај алгоритам би требало да убрза процес проналажења жељеног управљања, јер ће се мање пута десити прекид итеративног поступка услед мање ригорозних ограничења у поређењу са ВЕА алгоритмом.

2.2.1 COILC - доказ конвергенције алгорита

Посматра се систем дефинисан са (2.13), ограничен са

$$\mathbf{y}(t) = (\mathbf{y}_0(t), \mathbf{y}_1(t), \dots, \mathbf{y}_m(t)): \mathbf{y}_i(t) \in [\mathbf{Y}_i^{\max}, \mathbf{Y}_i^{\min}], i = 1, 2, \dots, m. \quad (2.42)$$

и подкуп остваривих жељених трајекторија дефинисаних са тачношћу праћења PLC методе μ :

$$Y_d = \{\mathbf{y}_d(t): \mathbf{y}_d(t) \in \mathbf{R}^m, \mathbf{y}_d(t) = (\mathbf{y}_1^d, \dots, \mathbf{y}_m^d), \mathbf{y}_i^d(t) \in [\mathbf{Y}_i^{\min} + \mu, \mathbf{Y}_i^{\max} - \mu], t \in [0, T], i = 1, 2, \dots, m\} \quad (2.43)$$

За систем дефинисан са (2.13) са ограничењима у простору стања дефинисаним са (2.42) и за сваку оствариву жељену трајекторију $\mathbf{y}_d \in Y_d$, ако се закон управљања (2.14) замени са

$$\begin{aligned} \mathbf{u}_{l+1}(t, S_l^s) &= (1 - \gamma)\mathbf{u}_l(t, S_{l-1}^s) + \gamma\mathbf{u}_0(t) + \mathbf{u}_l^*(t, S_l^s), \\ &\quad \mathbf{u}_0(t, S_{-1}^s) \equiv \mathbf{u}_0(t), \\ \mathbf{u}_l^*(t, S_l^s) &= \begin{cases} L(\mathbf{y}_l(t), t)(\dot{\mathbf{y}}_d(t) - \dot{\mathbf{y}}_l(t)), t \in [0, S_l^s], S_l^s \in (0, T]; \\ 0, \forall t \in (S_l^s, T] \end{cases} \end{aligned} \quad (2.44)$$

Теорема 1 и даље важи.

Доказ: Главна идеја доказа је да покаже да постоји секвенца

$$\{\|\delta\mathbf{u}_l(t, S_{l-1}^s)\|_{\lambda^*}, l \in \{0, \dots, \infty\}, t \in [0, T]\} \quad (2.45)$$

и да важи

$$\|\delta\mathbf{u}_{l+1}(t, S_l^s)\|_{\lambda^*} \leq \bar{\rho}\|\delta\mathbf{u}_l(t, S_{l-1}^s)\|_{\lambda^*} + \bar{\varepsilon}, 0 < \bar{\rho} < 1 \quad (2.46)$$

и стога, Лема 1 може бити примењена.

У случају $S_l^s \in (0, T)$ из (2.41) и (2.18) за $\|\delta\mathbf{u}_{l+1}(t, S_l^s)\|_{\lambda}$ и $\|\delta\mathbf{u}_l(t, S_{l-1}^s)\|_{\lambda}$

$$\sup_{t \in [0, T]} e^{-\lambda t} \|\delta\mathbf{u}_l(t, S_{l-1}^s)\| = \max \left(\begin{array}{l} \sup_{t \in [0, S_l^s]} e^{-\lambda t} \|\delta\mathbf{u}_l(t, S_{l-1}^s)\| \\ \sup_{t \in [S_l^s, T]} e^{-\lambda t} \|\delta\mathbf{u}_l(t, S_{l-1}^s)\| \end{array} \right) \quad (2.47)$$

и

$$\sup_{t \in [0, T]} e^{-\lambda t} \|\delta \mathbf{u}_{l+1}(t, S_l^s)\| = \max \left(\sup_{t \in [0, S_l^s]} e^{-\lambda t} \|\delta \mathbf{u}_{l+1}(t, S_l^s)\|, \sup_{t \in [S_l^s, T]} e^{-\lambda t} \|\delta \mathbf{u}_l(t, T_{S_{l-1}^s})\| \right) \quad (2.48)$$

Оба могућа случаја из (2.48) се разматрају за $\delta \mathbf{u}_{l+1}(t, S_l^s)$ када

$$\sup_{t \in [0, T]} e^{-\lambda t} \|\delta \mathbf{u}_{l+1}(t, S_l^s)\| = \sup_{t \in [S_l^s, T]} e^{-\lambda t} \|\delta \mathbf{u}_l(t, S_{l-1}^s)\| \quad (2.49)$$

и када

$$\sup_{t \in [0, T]} e^{-\lambda t} \|\delta \mathbf{u}_{l+1}(t, S_l^s)\| = \sup_{t \in [0, S_l^s]} e^{-\lambda t} \|\delta \mathbf{u}_{l+1}(t, S_l^s)\| \quad (2.50)$$

Применом хипер цилиндра полупречника ε око жељене трајекторије $\mathbf{y}_d(t)$ из (2.43) за \mathbf{u}_l из (2.44) и добије се $T_l: \|\mathbf{y}_l(T_l) - \mathbf{y}_d(T_l)\| = \varepsilon$ иако се покрет робота није зауставио када је $\|\mathbf{y}_l(T_l) - \mathbf{y}_d(T_l)\| = \varepsilon$, јер из (2.43) за било коју жељену излазну трајекторију $\mathbf{y}_d \in \mathbf{Y}_d$ постоји $\delta > 0: \mathbf{y}_d \in [\mathbf{Y}_i^{\min} + \varepsilon, \mathbf{Y}_i^{\max} + \varepsilon]$ и $\varepsilon = \mu + \delta > \mu$, и на основу тога $T_l \leq S_l$. На основу тога да Теорема 1 важи за \mathbf{u}_l из (2.44), $\varepsilon = \mu + \delta$ и $T_l \leq S_l$ и да постоји $k^* > 0$ такво да неједнакост (2.35) важи $\bar{\rho} \sup_{t \in [0, T_l]} e^{-k^* t} \|\delta \mathbf{u}_l(t, T_{l-1})\| \geq$

$\sup_{t \in [T_l, T]} e^{-k^* t} \|\delta \mathbf{u}_l(t, T_{l-1})\|$ и узимајући у обзир да $[0, T_l] \subset [0, S_l]$ и $[T_l, T] \supset [S_l, T]$, следи:

$$\bar{\rho} \sup_{t \in [0, S_l^s]} e^{-k^* t} \|\delta \mathbf{u}_l(t, S_{l-1}^s)\| \geq \sup_{t \in [S_l^s, T]} e^{-k^* t} \|\delta \mathbf{u}_l(t, S_{l-1}^s)\|. \quad (2.51)$$

Ако се примети да $0 < \bar{\rho} < 1$ из (2.47) и (2.51), следи:

$$\sup_{t \in [0, T]} e^{-k^* t} \|\delta \mathbf{u}_l(t, S_{l-1}^s)\| = \sup_{t \in [0, S_l^s]} e^{-k^* t} \|\delta \mathbf{u}_l(t, S_{l-1}^s)\|. \quad (2.52)$$

Узимајући у обзир да је $\bar{\varepsilon} \geq 0$ из неједнакости (2.51), комбинујући са (2.49) и (2.52) користећи (2.18) доводи до (2.45), и (2.46) са $\lambda^* = \max(k^*, \lambda)$.

За други случај важи неједнакост (2.50), Теорема 1 може директно бити примењена за $T > S_l: t \in [0, S_l]$ зато што су закони управљања (2.44) и (2.14) еквивалентни. Дакле, из (2.16) користећи (2.18) и узимајући у обзир да $\lambda(S_l) \leq \lambda(T)$, добија се:

$$\sup_{t \in [0, S_l]} e^{-\lambda t} \|\delta \mathbf{u}_{l+1}(t, S_l)\| \leq \bar{\rho} \left[\sup_{t \in [0, S_l]} e^{-\lambda t} \|\delta \mathbf{u}_l(t, S_{l-1})\| \right] + \bar{\varepsilon} \quad (2.53)$$

Комбиновањем једначина (2.50) и (2.52) са неједнакошћу (2.53), за $t \in [0, T]$ и $\lambda^* = \max(k^*, \lambda)$ добија се (2.45) и неједнакост (2.46).

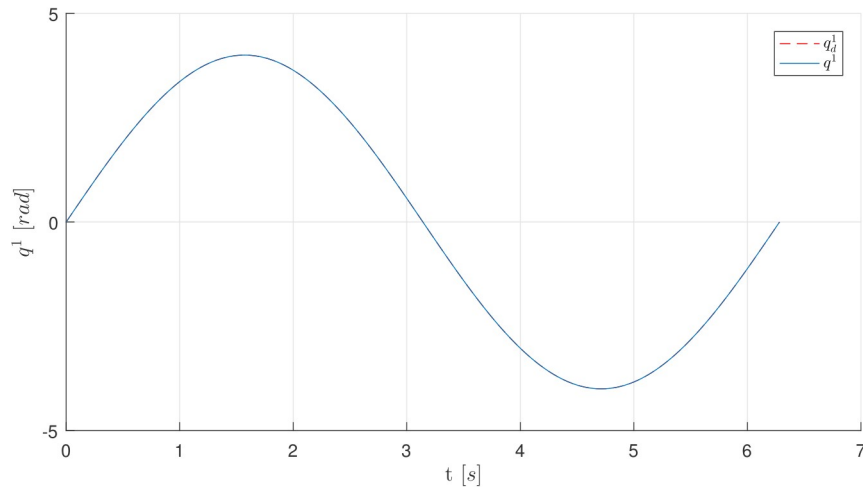
Под претпоставком да $l \rightarrow \infty$ може се применити Лема на (2.45) и (2.46), и добити неједнакости:

$$\begin{aligned} \limsup_{l \rightarrow \infty} \|\delta \mathbf{u}_l(t)\|_\lambda &\leq (1 - \bar{\rho})^{-1} \bar{\varepsilon} \\ \limsup_{l \rightarrow \infty} \|\delta \mathbf{y}_l(t)\|_\lambda &\leq b_y \end{aligned} \quad (2.54)$$

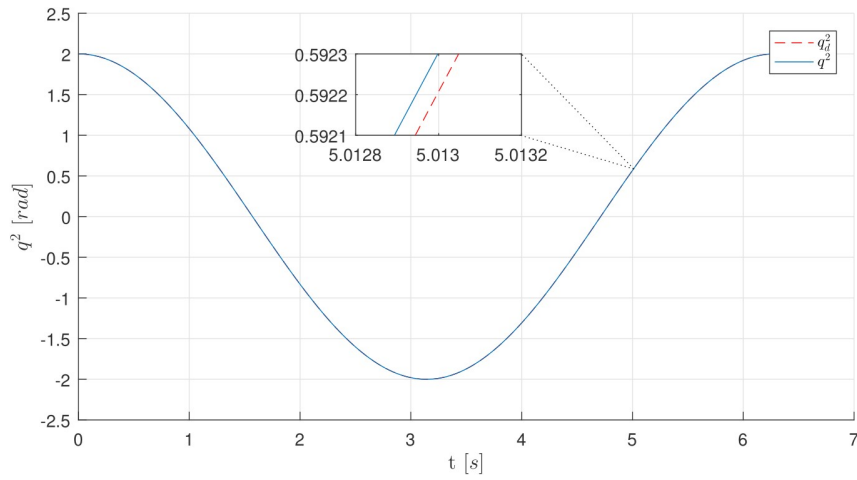
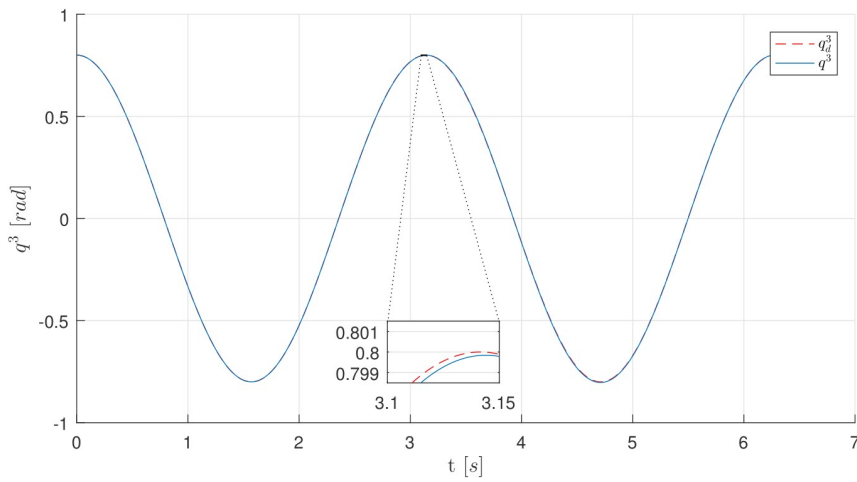
за нови закон управљања (2.44). Као последица тога, Теорема 1 и даље важи.

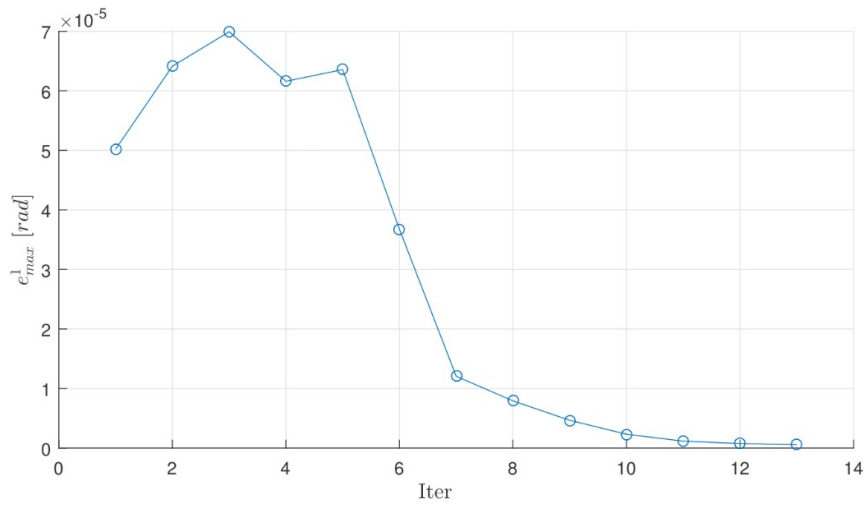
2.2.2 Пример управљања робота применом COILC алгоритма

Сада ће бити приказани резултати COILC алгоритма са истим сетом параметара (2.39), (Dubonjac, Lazarević, 2021):

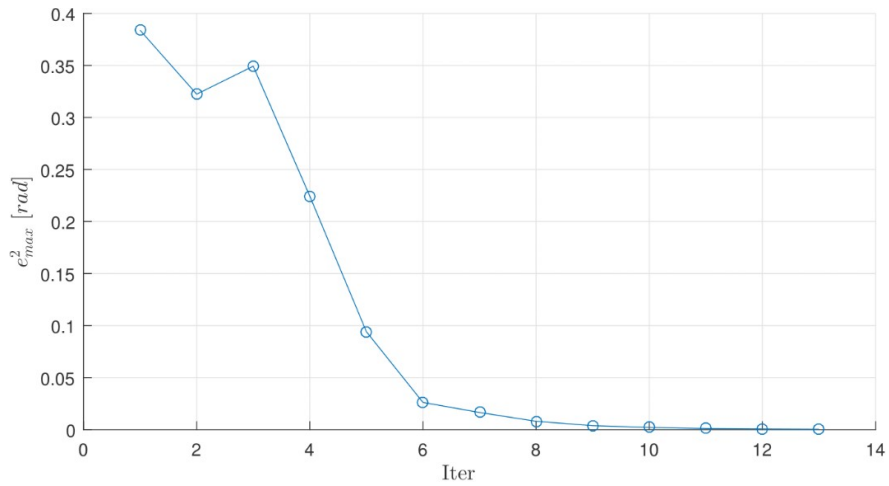


Сл. 2.14 Кретање генерализане координате q^1

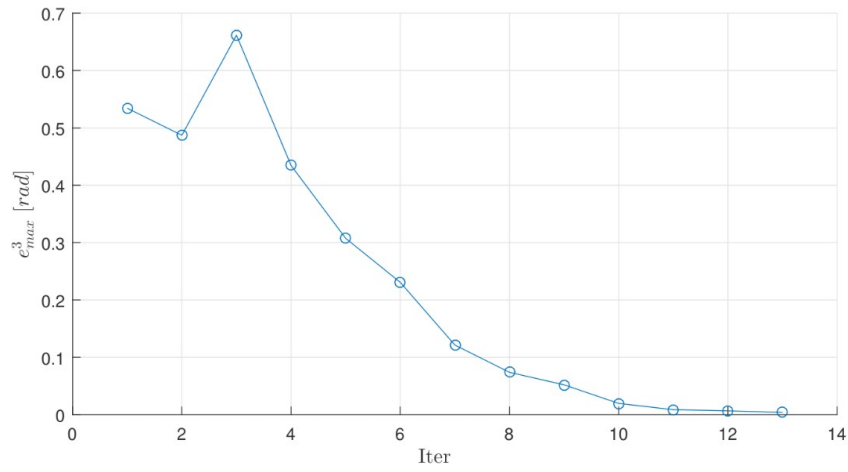
Сл. 2.15 Кретање генерализане координате q^2 Сл. 2.16 Кретање генерализане координате q^3



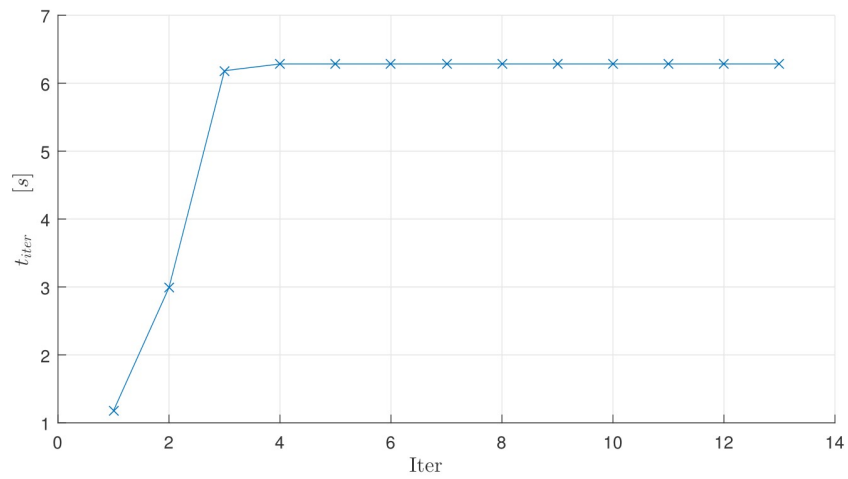
Сл. 2.17 Вредност норме максималне грешке праћења $\|e_{max}^1\|$ кроз итерације



Сл. 2.18 Вредност норме максималне грешке праћења $\|e_{max}^2\|$ кроз итерације



Сл. 2.19 Вредност норме максималне грешке праћења $\|e_{max}^3\|$ кроз итерације



Сл. 2.20 Време симулације по итерацијама

3. Напредни итеративни алгоритам управљања учењем за рехабилитационе егзоскелетоне

3.1 Увод у рехабилитациону роботску

Концепт рехабилитационе роботике представља једно од најважнијих питања у социјално роботским истраживањима. Представљен је 60-тих година као помоћно средство у рехабилитацији након шлога, и од тада је представљен велики број машина које ће бити употребљиване за ту сврху. Данас постоје два типа рехабилитационих робота (*Mehdi and Boubaker, 2012*):

- Рехабилитациони роботи горњих екстремитета
- Рехабилитациони роботи доњих екстремитета

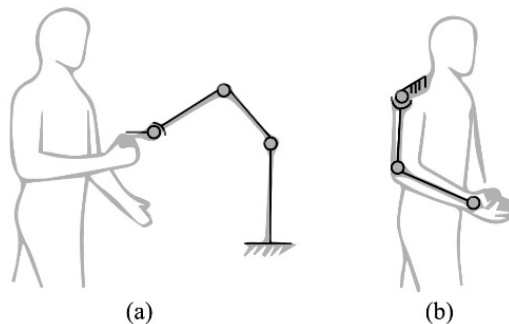
У овом доприносу, акценат ће бити стављен на *прву групу* рехабилитационих робота. Рехабилитација пацијента, односно терапија која се пружа пацијентима након шлога, може бити потребна и дужи временски период од „предвиђеног“, што за пацијенте може представљати додатни напор, а са друге стране ни финансијски аспекти нису погодни за клинике или рехабилитационе установе. Многи фактори који доприносе бржем повратку моторних способности пацијената се не искоришћавају у потпуности услед недостатка рехабилитационе услуге. Оно што је утврђено, јесте да рехабилитација даје најбоље резултате ако се примењује одмах након шлога. Нажалост то није увек могуће урадити. Студије су показале да интензивна терапија, као и вежбе које су засноване на унапред одређеном задатку, доприносе бољем опоравку тј. боље утичу на повратак моторних функција пацијената. Физиотерапија, за пацијенте који су преживели шлог, указује на то да се опоравак пацијената може наставити и дуги низ година након шлога (*Losseff, 2004*). Међутим, сматра се да ће са повећањем броја пацијената који доживе шлог, а тај број је у сталном порасту, пружање адекватног третмана бити јако отежано имајући у виду природу интензивног рада који мора бити уложен. Управо роботи (егзоскелетони) имају потенцијала да задовоље велику потражњу, са којом конвенционална терапија не може да се избори. Ови роботи су дизајнирани тако да имају сличну кинематичку структуру као и људски уд. Још једна предност се огледа и у могућности да ови роботи могу бити ношени од стране пацијената.

У поређењу са конвенционалном терапијом, егзоскелети пружају могућност интензивне терапије дужи временски период (неколико година након шлога). Предност је и то што су роботи потпуно независни од људског фактора (вештина или замор терапеута). Роботи омогућавају терапију без присуства терапеута, што даље може водити до чешћих терапија а и смањивања трошкова. Осим тога, роботи пружају

и могућност мерења квантитативних података, на основу којих је могуће доносити закључке о стању пацијента. Данас се дизајнирају и виртуелне игре које за циљ имају да терапију учине забавнијом за пацијента, као и да подстакну пацијента да уложи више труда у терапију.

3.1.1 Рехабилитациони роботи горњих екстремитета

Ранија истраживања везана рехабилитациону роботску терапију горњих екстремитета била су базирана на принципу *end-effector* робота. *End-effector* роботи придржавају руку пацијента или његову подлактицу (у једној тачки) и генеришу силе на интерфејсу (Слика 3.1 а) (Bouteraa, Abdallah, 2016).



Слика 3.1 а) *end-effector* робот б) егзоскелетон

Зглобови *end-effector* робота не oponашају зглобове човека. Овај тип робота је једноставнији, лакши се израђује и може се лако подешавати како би одговарало било којој дужини горњих екстремитета. Међутим одређивање положаја горњих екстремитета на основу само једног интерфејса може бити веома тешко, поготово када се тај интерфејс налази у човековој руци. Контролисање обртних момената појединих зглобова представља немогућ задатак, што за резултат има неконтролисан пренос оптерећења између зглобова горњих екстремитета. Као последица јавља се потешкоћа у генерисању појединачног покрета једног зглоба горњег екстремитета, јер кретање крајњег ефектора може изазвати комбиноване покрете зглобова шаке, лакта и рамена. Поред тога опсег покрета код оваквог типа робота је ограничен, чиме се самим тим и ограничава број рехабилитационих вежби који се могу извести (Anam and Al-Jumaily, 2012).

Новије генерације робота које су у употреби, и којима се посвећује више пажње јесу егзоскелетон роботи. Оно што овај тип робота чини погоднијим за употребу и истраживање, јесте то што поседују структуру која јако подсећа на структуру горњих

екстремитета човека. Такође, осе зглобова овог типа робота подударају се са осама зглобова човека. Егзоскелетони су дизајнирани тако да раде у складу са горњим екстремитетима, и стога могу бити везани за већи број карактеристичних места горњих екстремитета. Иако ово доводи до тога да је доста теже прилагодити робота различитим дужинама екстремитета човека, већи број интерфејса омогућава да се у потпуности одреди положај руке и такође, да се контролише обртни момент сваког зглоба понаособ. У односу на *end-effector* тип робота, егзоскелетон поседује шири опсег покрета, и самим тим омогућава већи број рехабилитационих вежби које се могу применити.

3.2 Биомеханички објекат управљања

У следећим поглављима разматра се систем човек – машина, који се састоји од роботског система (егзоскелетон) са три степена слободе и људска рука са такође три степена слободе (три ротације). Људска рука се састоји из два сегмента – подлактица и надлактица, и два зглоба – рамени зглоб и лакат. Рамени зглоб поседује два степена слободе, док лакат поседује један степен слободе. Роботски систем (егзоскелетон) има такође два сегмента, два зглоба између сегмената од којих рамени зглоб поседује два, а лакат један степен слободе.

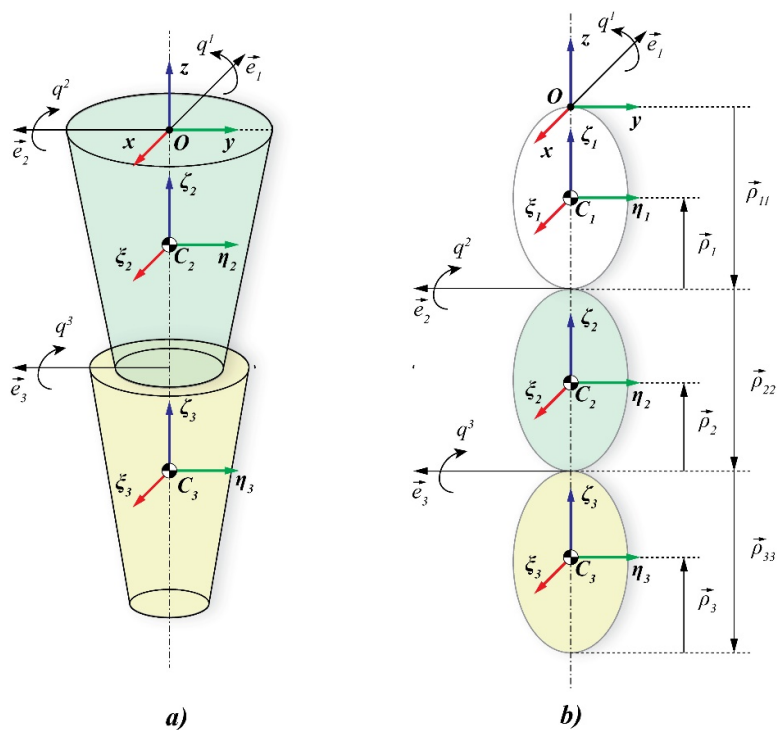
Комбиновањем ова два система добија се један поједностављен биомеханички систем. Сматра се да тако изведени систем има два сегмента, два зглоба и три степена слободе. Рамени зглоб поседује два степена слободе, а лакат један степен слободе. Сва три степена слободе су ротације. За облик сегмената су узете зарубљене купе (*Velandia, Tibaduiza and Vejar, 2017*). Поред наведених поједностављења уведе се и следеће претпоставке.

- Претпоставка 1. Сви сегменти представљају крута тела.
- Претпоставка 2. Материјал од кога је робот направљен и ткиво људске руке се сматра хомогеним.
- Претпоставка 3. Занемарује се утицај трења клизања, вискозног трења, тј. одговарајућих сила трења, вискозних сила.
- Претпоставка 4. Све везе у датом биомеханичком систему су холономне (геометријске). Овакве везе ограничавају само померање тачака система.

Усвајање, израчунавање и извођење математичког модела система врши се поступцима као и наведеним књигама (*Човић, Лазаревић, 2009*), (*Лазаревић, 2006*).

3.2.1 Кинематичке и динамичке карактеристике биомеханичког система

Дати биомеханички систем поседује, као што је већ речено, два сегмента и два зглоба која омогућавају кретање сегмената. Рамени зглоб има два степена слободe, што значи представља кинематички пар четврте класе. Овакав зглоб могуће је декомпоновати на два кинематичка пара пете класе (један степен слободe) и при томе се уводи један фиктивни сегмент. Маса и дужина фиктивног сегмента је једнака нули. Лакат представља кинематички пар пете класе и није га потребно декомпоновати. Након декомпозиције усваја се непокретни (референтни) координатни систем $Oxyz$ као што је приказано на слици (Слика 3.2 б)). Затим се сваком сегменту (V_i) придружује локални координатни систем $C_i\xi_i\eta_i\zeta_i$, где је C_i тачка центар масе сегмента. Ови системи су круто везани за сегменте и крећу се заједно са њима. У почетном тренутку осе свих локалних координатних система се поклапају са осама непокретног координатног система и то се назива референтна конфигурација.



Слика 3.2 а) Шематски приказ биомеханичког система, б) Шематски приказ декомпонованог биомеханичког система

Основни подаци за дати биомеханички систем су дати у следећој табели:

Сегмент i	1	2	3
Дужина $l_i[m]$	0	0.369	0.36
Полупречник $r_i[m]$	0	0.04	0.01
Полупречник $R_i[m]$	0	0.09	0.05
Маса $m_i[kg]$	0	1.4	1.1

На основу усвојене референтне конфигурације и основних података из претходне табеле могу се израчунати вредности јединичних вектора, вектора положаја и тензора инерције потребних за извођење диференцијалних једначина кретања. У случају датог биомеханичког система јединични вектори, вектори положаја су дати у следећој табели:

i	1	2	3
$\{e_i^{(i)}\} [m]$	$[-1 \ 0 \ 0]^T$	$[0 \ -1 \ 0]^T$	$[0 \ -1 \ 0]^T$
$\{\rho_{ii}^{(i)}\} [m]$	$[0 \ 0 \ 0]^T$	$[0 \ 0 \ -0.369]^T$	$[0 \ 0 \ -0.36]^T$
$\{\rho_i^{(i)}\} [m]$	$[0 \ 0 \ 0]^T$	$[0 \ 0 \ 0.2296]^T$	$[0 \ 0 \ 0.2497]^T$

Тензори инерције за дати систем су:

$$\{J_{c1}^{(1)}\} = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \quad (3.1)$$

$$\{J_{c2}^{(2)}\} = \begin{bmatrix} 0.03 & 0 & 0 \\ 0 & 0.03 & 0 \\ 0 & 0 & 0.0037 \end{bmatrix} \quad (3.2)$$

$$\{J_{c3}^{(3)}\} = \begin{bmatrix} 0.0218 & 0 & 0 \\ 0 & 0.0218 & 0 \\ 0 & 0 & 0.0008 \end{bmatrix}. \quad (3.3)$$

3.2.2 Диференцијалне једначине кретања

Диференцијалне једначине кретања за дати биомеханички систем се одређују у облику Лагранжевих једначина друге врсте:

$$\frac{d}{dt} \frac{\partial E_k}{\partial \dot{q}_\gamma} - \frac{\partial E_k}{\partial q_\gamma} = Q_\gamma, \gamma = 1, 2, \dots, n \quad (3.4)$$

где је Q_γ генералисана сила система активних сила које делују на дати систем, која одговара независној генералисаној координати q_γ . Одређивање диференцијалних једначина кретања у облику отвореног кинематичког ланца своди се на одређивање величина које фигуришу у изразу (3.4). Да би се искористила предност одређивања једначина кретања на основу метричког тензора, потребно је израз (3.4) написати у другачијем облику. На основу извођења, које је детаљно објашњено у (Човић и Лазаревић, (2007)), имамо да важе следеће једнакости:

$$\frac{d}{dt} \left(\frac{\partial E_k}{\partial \dot{q}_\gamma} \right) = \sum_{\alpha=1}^n a_{\alpha\gamma} \ddot{q}^\alpha + \frac{1}{2} \sum_{\alpha=1}^n \sum_{\beta=1}^n \left(\frac{\partial a_{\alpha\gamma}}{\partial q^\beta} + \frac{\partial a_{\beta\gamma}}{\partial q^\alpha} \right) \dot{q}^\alpha \dot{q}^\beta \quad (3.5)$$

$$\frac{\partial E_k}{\partial q^\gamma} = \frac{1}{2} \sum_{\alpha=1}^n \sum_{\beta=1}^n \frac{\partial a_{\alpha\beta}}{\partial q^\gamma} \dot{q}^\alpha \dot{q}^\beta. \quad (3.6)$$

На основу израза (3.5) и (3.6) следи нови облик једначина (3.4):

$$\sum_{\alpha=1}^n a_{\alpha\gamma} \ddot{q}^\alpha + \sum_{\alpha=1}^n \sum_{\beta=1}^n \Gamma_{\alpha\beta,\gamma} \dot{q}^\alpha \dot{q}^\beta = Q_\gamma, \gamma = 1, \dots, n. \quad (3.7)$$

где су $a_{\alpha\gamma}$ метрички тензор, а $\Gamma_{\alpha\beta,\gamma}$ Кристофелови симболи прве врсте. Овакав облик једначина кретања се назива коваријантним. На основу усвојених и израчунатих кинематских и динамичких карактеристика система из претходног дела може се израчунати метрички тензор, а на основу њега и Кристофелови симболи прве врсте за дати систем. За детаљан поступак израчунавања видети (Човић, Лазаревић, 2009). На основу тога сада се могу израчунати диференцијалне једначине кретања у функцији генералисаних координата датог система и оне гласе:

$$\begin{aligned} &0.0217\dot{q}^1 + 0.595 \sin(q^1 + q^2 + q^3) + 2.95 \sin(q^1 - q^2) + 2.95 \sin(q^1 + q^2) \\ &\quad - 0.595 \sin(q^2 - q^1 + q^3) + 0.204\dot{q}^1 (\cos q^2)^2 \\ &\quad + 0.0448\dot{q}^1 \cos(2q^2 + q^3) + 0.0448\dot{q}^1 \cos q^3 \\ &\quad + 0.0172\dot{q}^1 \cos(2q^2 + 2q^3) - 0.0448\dot{q}^1 \dot{q}^3 \sin q^3 \\ &\quad - 0.0344\dot{q}^1 \dot{q}^2 \sin(2q^2 + 2q^3) - 0.0344\dot{q}^1 \dot{q}^3 \sin(2q^2 + 2q^3) \\ &\quad - 0.0896\dot{q}^1 \dot{q}^2 \sin(2q^2 + q^3) - 0.0448\dot{q}^1 \dot{q}^3 \sin(2q^2 + q^3) \\ &\quad - 0.204\dot{q}^1 \dot{q}^2 \sin 2q^2 = Q_1 \end{aligned} \quad (3.8)$$

$$\begin{aligned}
& 0.242\ddot{q}^2 + 0.0352\ddot{q}^3 - 0.0448(\dot{q}^3)^2 \sin q^3 + 5.9 \cos q^1 \sin q^2 \\
& \quad + 0.0172(\dot{q}^1)^2 \sin(2q^2 + 2q^3) + 0.0448(\dot{q}^1)^2 \sin(2q^2 + q^3) \\
& \quad + 0.0896\dot{q}^2 \cos q^3 + 0.0448\dot{q}^3 \cos q^3 + 0.102(\dot{q}^1)^2 \sin 2q^2 \\
& \quad - 0.0896\dot{q}^2 \dot{q}^3 + 1.19 \cos q^1 \cos q^2 \sin q^3 \\
& \quad + 1.19 \cos q^1 \cos q^3 \sin q^2 = Q_2
\end{aligned} \tag{3.9}$$

$$\begin{aligned}
& [0.0224 \sin(2q^2 + q^3) + 0.0224 \sin q^3 + 0.0172 \sin(2q^2 + 2q^3)](\dot{q}^1)^2 \\
& \quad + 0.0448 \sin q^3 (\dot{q}^2)^2 + 0.0352\dot{q}^3 + \ddot{q}^2(0.0448 \cos q^3 + 0.0352) \\
& \quad + 1.19 \cos q^1 \cos q^2 \sin q^3 + 1.19 \cos q^1 \cos q^3 \sin q^2 = Q_3
\end{aligned} \tag{3.10}$$

3.3 Увод у управљање роботског система

Роботски систем представља један *нелинеарни вишеструко преносни нестационарни динамички систем* (Човић, Лазаревић, 2007). Структура једног оваквог система се састоји из управљачког система и објекта управљања. Објекат управљања подразумева роботски механизам са актуаторима, односно у датом случају егзоскелетни механизам и људска рука. Управљачки систем је најчешће дигиталног типа, тј. представља један дискретан систем, међутим овде ће бити представљен управљачки систем континуалног типа који ће послужити као сасвим добра апроксимација. Задатак управљања може се дефинисати као: „*Неопходно је обезбедити такву промену управљачких величина тако да завршни уређај роботског система оствари захтевано кретање у простору*“ (Човић, Лазаревић, 2009).

3.3.1 Управљање у простору стања и у простору излаза

Управљање роботским системом (биомеханички систем у даљем тексту) дели се на управљање у *простору стања* и управљање у *простору излаза*. Простор стања се код роботских система другачије назива *простор зглобова*, а простор излаза *операциони простор*. У овом раду је приказано управљање у простору зглобова и управљање у операционом простору није детаљније разматрано (Лазаревић, Баталов и Живановић, 2014).

једначине кретања су диференцијалне једначине другог реда. Преласком у простор стања врши се редукција реда система и уместо система од n диференцијалних једначина другог реда добија се систем од $2n$ диференцијалних једначина првог реда. Овако добијене једначине су у општем случају зависне.

Прелазак у простор стања се врши на следећи начин. Нека је $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{x} = [\mathbf{x}_p, \mathbf{x}_v]^T$ вектор стања и нека је $\mathbf{u} = \mathbf{Q}(\mathbf{q}, \dot{\mathbf{q}}, t)$ вектор управљања. Уводи се смена променљивих тако да важи:

$$\mathbf{x}_p = [x_1 \ x_2 \ x_3]^T = [q^1 \ q^2 \ q^3]^T, \mathbf{x}_v = [x_4 \ x_5 \ x_6]^T = [\dot{q}^1 \ \dot{q}^2 \ \dot{q}^3]^T \quad (3.26)$$

Сада, диференцијалне једначине кретања могу се написати у следећем облику:

$$\dot{\mathbf{x}} = A(\mathbf{x}) + B(\mathbf{x})\mathbf{u} \quad (3.27)$$

где су

$$A(\mathbf{x}) = \begin{bmatrix} \mathbf{x}_v \\ -[a(\mathbf{x}_p)]^{-1} \mathbf{c}(\mathbf{x}_p, \mathbf{x}_v) \end{bmatrix}, B(\mathbf{x}) = \begin{bmatrix} 0_{n \times n} \\ [a(\mathbf{x}_p)]^{-1} \end{bmatrix} \quad (3.28)$$

Овај облик представља векторску једначину стања датог биомеханичког система. Ако се уведе и вектор излаза $\mathbf{y} = [\mathbf{y}_p, \mathbf{y}_v]^T = [\bar{\mathbf{q}}, \dot{\bar{\mathbf{q}}}]^T$ могуће је дефинисати и једначину излаза биомеханичког система. Једначина излаза има општи облик:

$$\mathbf{y} = C(\mathbf{x}), C(\mathbf{x}) = [f(\mathbf{q}) \quad [J(\mathbf{q})]\dot{\mathbf{q}}]^T \quad (3.29)$$

Као што је раније напоменуто, референтне трајекторије су формиране директно у простору зглобова (простор стања), тако да нема потребе за израчунавањем инверзног задатка кинематике и стога једначина излаза може бити поједностављена:

$$\mathbf{y} = \mathbf{x} = [\mathbf{x}_p \ \mathbf{x}_v]^T \quad (3.30)$$

Биомеханички систем који је до сада разматран је један нелинеаран систем, као што је већ речено. Самим тим потребно је применити нелинеарне законе при дизајну управљачког система. Идеја *Feedback* линеаризације је да се путем повратне затворене петље добије линеарна зависност улаза и излаза. Другим речима све нелинеарне величине које фигуришу у једначинама кретања компензују се одговарајућим законом управљања. Треба напоменути да је *Feedback* линеаризација егзактна линеаризација, тј. разликује се од јакобијанске линеаризације која је линеарна апроксимација динамике датог система Slotine, (1991).

Feedback линеаризација је општи концепт управљања које је у литератури о роботизи познатије као *Computed torque*. Пошто је концепт *Computed torque* интуитивнији биће примењен на дати систем.

Креће се од једначине биомеханичког система

$$a(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{c}(\mathbf{q}, \dot{\mathbf{q}}) = \mathbf{Q}^u \quad (3.31)$$

Где је $\mathbf{c}(\mathbf{q}, \dot{\mathbf{q}}) = b(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} - \mathbf{Q}^g(\mathbf{q})$, $a(\mathbf{q})$ је основни метрички тензор система и \mathbf{Q}^u вектор генералисаних погонских, односно управљачких сила.

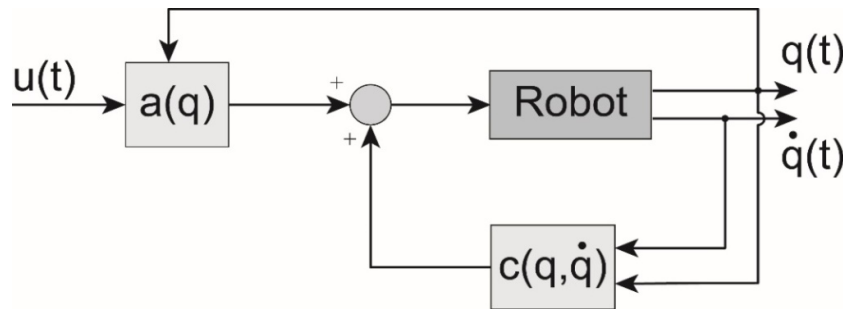
Закон управљања који ће неутралисати нелинеарности система има облик:

$$\mathbf{Q}^u = a(\mathbf{q})\mathbf{v} + \mathbf{c}(\mathbf{q}, \dot{\mathbf{q}}) \quad (3.32)$$

Уврштавањем претходног израза у једначину система (3.31), добија се линеарни декупловани систем у облику двоструког интегратора

$$\ddot{\mathbf{q}} = \mathbf{v} \quad (3.33)$$

Систем има линеарну везу улаза и излаза. Тиме је затворена петља која врши егзактну линеаризацију и сада је могуће применити неки од линеарних закона управљања да би се постигло жељено понашање система. Из претходног се види да тренутним израчунавањем генералисаних сила момената (одатле назив *Computed torque*) на излазу и враћањем тих вредности повратном петљом, постиже се линеаризација. У случају немоделоване динамике или било каквих неизвесности у динамици, губи се на робусности система управљања. Изглед блок дијаграма система који је егзактно линеаризован је дат на следећој слици (Слика 3.9).



Слика 3.9. Блок дијаграм система са егзактном (*feedback*) линеаризацијом

3.3.4 Синтеза линеарног закона управљања-алгоритам управљања итеративним учењем

Након линеаризације система сада је могуће одабрати линеарни закон управљања који ће довести понашање система у жељене оквире. На систем ће бити примењени закони управљања и у директној и повратној грани. У повратној грани примењен је класичан *PD* закон управљања, а у директној грани закон управљања итеративним учењем. У једначине кретања уводи се додатни параметар неизвесности. Пошто се егзоскелетни

систем састоји од роботске и људске руке није могуће прецизно моделовати људску руку, тако да ће увек постојати одређена неизвесност у моделу. Нова једначина система гласи:

$$\dot{\mathbf{x}} = \begin{Bmatrix} \mathbf{x}_v \\ \mathbf{v} \end{Bmatrix} + \boldsymbol{\eta}(t) \quad (3.34)$$

Неизвесност $\boldsymbol{\eta}(t)$ је додата на два начина у линеаризовани модел. У првом случају неизвесност је једноставно усвојена као синусна односно косинусна функција.

$$\boldsymbol{\eta}_1(t) = \begin{Bmatrix} 0.1 \cos(2\pi t) \\ 0.5 \cos(\pi t) \\ 0.2 \sin(\pi t) \end{Bmatrix} \quad (3.35)$$

У другом случају је узето да се матрице система (3.31) $a(\mathbf{q})$, $\mathbf{c}(\mathbf{q}, \dot{\mathbf{q}})$ могу поделити на номинални и неизвесни део *Lazarević, et al. (2020)*. Тада се систем једначина (3.31) може написати у следећем облику:

$$(a_N(\mathbf{q}) + \Delta a)\ddot{\mathbf{q}} + (B_N(\mathbf{q}, \dot{\mathbf{q}}) + \Delta B) - (\mathbf{Q}_N^g + \Delta \mathbf{Q}^g) = \mathbf{Q}^m \quad (3.36)$$

Ако се узме да је:

$$\mathbf{Q}^m = a_N(\mathbf{q})\mathbf{u} + B_N(\mathbf{q}, \dot{\mathbf{q}}) - \mathbf{Q}_N^g \quad (3.37)$$

тада следи да је:

$$\ddot{\mathbf{q}}(t) = a^{-1}(\mathbf{q})a_N(\mathbf{q})\mathbf{u}(t) + \boldsymbol{\eta}(\mathbf{q}, \dot{\mathbf{q}}) \quad (3.38)$$

где је

$$\boldsymbol{\eta}_2(\mathbf{q}, \dot{\mathbf{q}}) = -a^{-1}(\mathbf{q})(\Delta B(\mathbf{q}, \dot{\mathbf{q}}) - \Delta \mathbf{Q}^g) \quad (3.39)$$

Следи детаљнији опис закона управљања у повратној и директној грани система.

Управљање путем итеративног учења је првобитно нашло примену у индустријским аутоматизованим системима где систем врши исту операцију много пута, током непромењеног коначног временског интервала. По завршетку сваке операције, систем се ресетује и поново доводи у почетни положај, одакле поново врши исту операцију. Уобичајен пример оваквог понашања је робот који премешта објекте са једног места на друго.

Управљање путем итеративног учења се разликује од класичних метода управљања по томе што користи управљачке информације из претходних вршења операције, као што су грешка и улазни сигнал, при конструкцији контролног улазног сигнала за тренутно вршење задатка. Постоје две фазе у примени управљања итеративним учењем (ILC - *Iterative Learning Control*, превод са енглеског језика): прво се похрањују информације о контролном улазном сигналу из прошле итерације у меморију, а потом се те информације користе у неком облику као контролни улазни сигнал у тренутној

итерацији који ће обезбедити конвергенцију грешке, робусност система итд (*Arimoto, et al. 1984*).

Проблем управљања итеративним учењем захтева увођење одређених аксиома:

- Свака итерација се дешава у ограниченом временском интервалу $t \in [0, T]$.
- Жељена излазна трајекторија је дата *a priori* за тај временски интервал трајања итерације $t \in [0, T]$.
- Понављање почетних услова из итерације у итерацију је задовољено, тј. почетно стање $x_k(0)$ система је увек исто на почетку сваке итерације $x_k(0) = x_0$, $k = 1, 2, \dots$.
- Инваријантност динамике система је обезбеђена током поновљених итерација.
- Сваки излаз u_k може бити искоришћен за конструкцију следећег улаза $u_{k+1}(t)$.
- Динамика система је инвертибилна, тј. за дат жељени излаз $y_d(t)$ са непрекидним изводом, постоји јединствен улаз $u_d(t)$ који побуђује систем и даје излаз $y_d(t)$.

Потребно је остварити закон управљања такав да обезбеђује да излаз система $y(t)$ конвергира унапред задатој референтној трајекторији $y_d(t)$ за коначни временски интервал $t \in [0, T]$.

Основни облик закона управљања код итеративног учења има облик:

$$\mathbf{u}_{k+1}(t) = \mathbf{u}_k(t) + L\dot{\mathbf{e}}_k(t) \quad (3.40)$$

где је L појачање, а k је број итерација.

У овом раду ће бити примењен мало другачији закон учења чије коришћење може бити оправдано мање захтевним рачунаром. Нека је позициона грешка за сваку итерацију дефинисана као

$$\mathbf{e}_k(t) = \mathbf{y}_d(t) - \mathbf{y}_k(t) \quad (3.41)$$

и нека је se_k сума грешака за сваку итерацију током времена t :

$$se_k = \sum_0^T \mathbf{e}_k(t) \quad (3.42)$$

и нека је

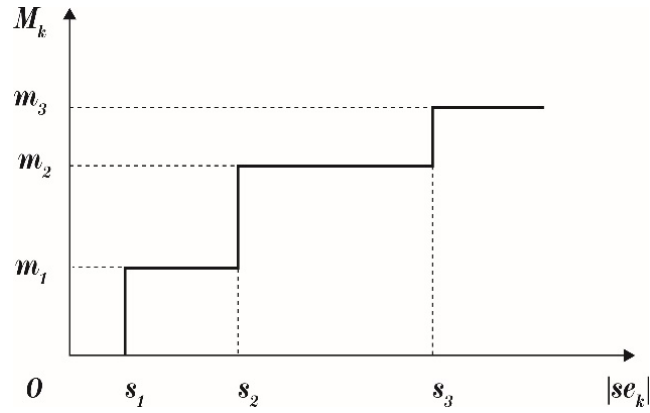
$$M_k = f(|se_k|) \quad (3.43)$$

Функција се може усвојити да буде једноставна степ функција због смањења потребног места за складиштење у меморији. Принцип функције је приказан на слици (Слика 3.10). Као што се са слике може видети за велику суму грешке M_k има сразмерно велику вредност и обрнуто.

Облик закона управљања је комбинација закона управљања преузетих из (*Wang, 2012*) и (*Lazarević, et al. 2020*) и он гласи:

$$\mathbf{u}_{k+1}^{ff}(t) = \mathbf{u}_k(t) + \text{sgn}(\mathbf{e}_k(t))M_k + \text{sgn}(\dot{\mathbf{e}}_k(t))M'_k + \text{sgn}(\ddot{\mathbf{e}}_k(t))M''_k \quad (3.44)$$

где је \mathbf{e}_k вектор грешке позиције, $\dot{\mathbf{e}}_k$ је вектор грешке брзине, $\ddot{\mathbf{e}}_k$ је вектор грешке убрзања, а M'_k и M''_k су функције аналогне претходно објашњеној степ функцији M_k . Једначина (3.44) представља члан закона управљања у директној грани тј. *feedforward* члан.



Слика 3.10 Степ функција у зависности од суме грешака

ПД алгоритмом управљања може се успешно вршити управљање позицијом роботског система чији модел не садржи гравитациони члан (Лазаревић, 2005). У нашем моделу постоји гравитациони члан, међутим он је компензован затварањем унутрашње нелинеарне петље. У том случају процес подешавања параметара управљања је једноставан и захтева само да матрице K_p и K_v буду симетричне, позитивно одређене матрице (Лазаревић, Баталов и Живановић, 2014). Закон управљања у повратној спрези за дати систем гласи (Lazarević i Živković, 2021):

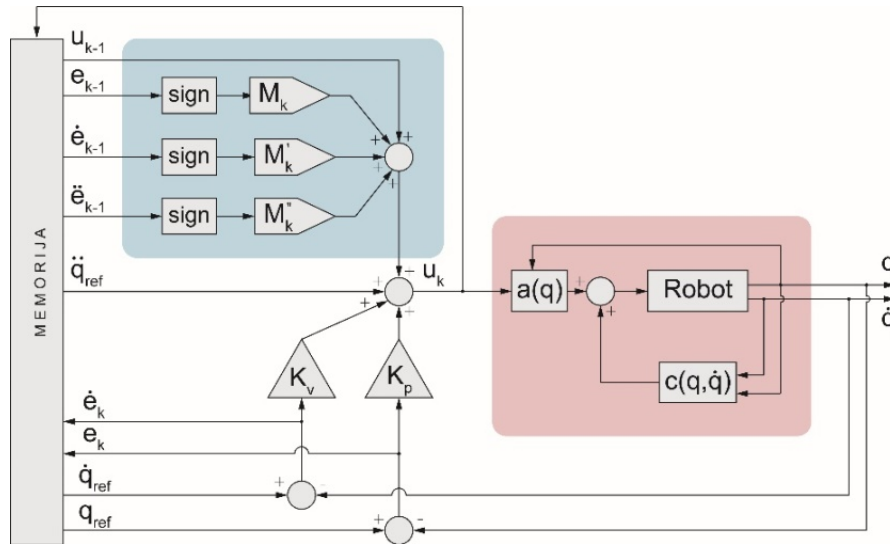
$$\mathbf{u}_{k+1}^{fb}(t) = K_p \mathbf{e}_{k+1}(t) + K_v \dot{\mathbf{e}}_{k+1}(t) \quad (3.45)$$

Сада можемо написати коначни закон управљања комбиновањем једначина (3.44) и (3.45):

$$\mathbf{u}_{k+1} = \mathbf{u}_{k+1}^{ff}(t) + \mathbf{u}_{k+1}^{fb}(t) \quad (3.46)$$

односно

$$\mathbf{u}_{k+1} = \mathbf{u}_k(t) + \text{sgn}(\mathbf{e}_k(t))M_k + \text{sgn}(\dot{\mathbf{e}}_k(t))M'_k + \text{sgn}(\ddot{\mathbf{e}}_k(t))M''_k + K_p \mathbf{e}_{k+1}(t) + K_v \dot{\mathbf{e}}_{k+1}(t) \quad (3.47)$$



Слика 3.11 Блок дијаграм целокупног система

Блок дијаграм сада када је затворена и спољашња петља управљања можемо представити као што је приказано на слици (Слика 3.11). Након успостављања једначине улаза може се прећи на симулацију и дискусију добијених резултата, што и следи у даљем излагању.

3.3.5 Резултати симулације

За потребе нумеричке симулације користи се линеаризовани модел у простору стања са придодатим чланом који представља неизвесности у моделу. Примењен је закон управљања претходно изведен. Као жељене трајекторије користе се, такође претходно израчунате, трајекторије у облику полинома петог реда. Нека је потребно задовољити услов да позициона грешка задовољава услов $|\varepsilon| \leq 0.005[\text{rad}]$.

Први случај - $\eta = \eta_1(t)$

Параметри управљања за први случај су следећи:

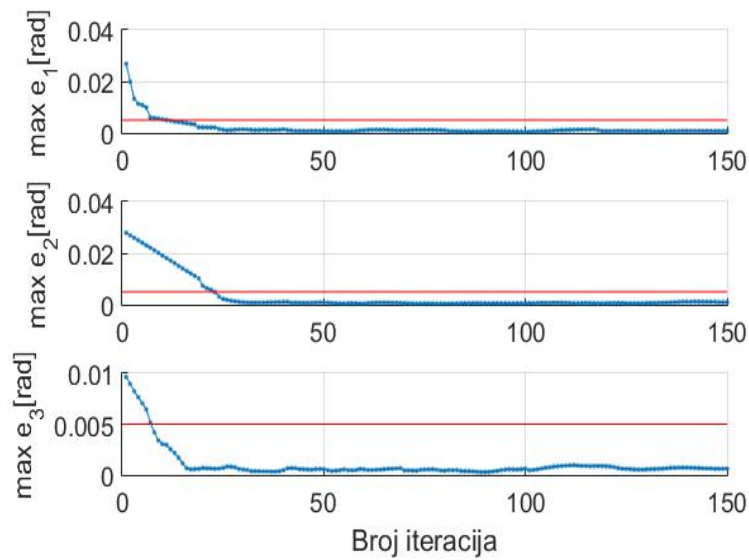
$$K_p = \begin{bmatrix} 15 & 0 & 0 \\ 0 & 15 & 0 \\ 0 & 0 & 15 \end{bmatrix}, K_v = \begin{bmatrix} 6 & 0 & 0 \\ 0 & 6 & 0 \\ 0 & 0 & 6 \end{bmatrix} \quad (3.48)$$

$$M_{k1} = M_{k2} = M_{k3} = \begin{cases} 0.09, & |se_k| > 0.5 \\ 0.005, & 0.1 < |se_k| \leq 0.5 \\ 0.001, & |se_k| \leq 0.1 \end{cases} \quad (3.49)$$

$$M'_{k1} = M'_{k2} = M'_{k3} = \begin{cases} 0.09, & |se_k| > 0.2 \\ 0.05, & 0.1 < |se_k| \leq 0.2 \\ 0.01, & |se_k| \leq 0.1 \end{cases} \quad (3.50)$$

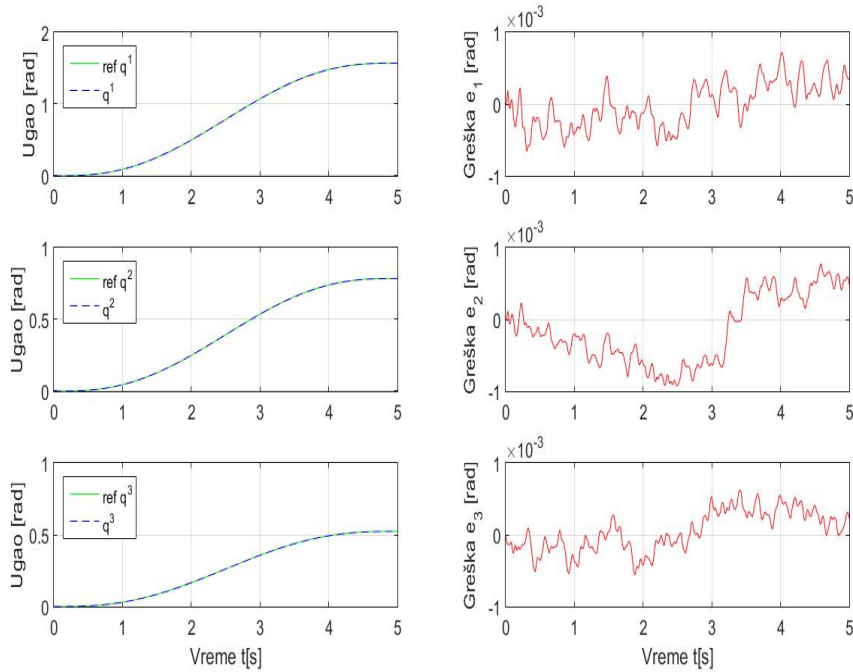
$$M''_{k1} = M''_{k2} = M''_{k3} = \begin{cases} 0.01, & |se_k| > 1 \\ 0.005, & 0.5 < |se_k| \leq 1 \\ 0.001, & |se_k| \leq 0.5 \end{cases} \quad (3.51)$$

Параметри су одабрани на основу простих покушаја и грешки. На следећим сликама су приказане максималне апсолутне вредности грешака током итерација.



Слика 3.12 Максималне позиционе грешке по итерацијама

Са слике (Слика 3.12) се може уочити конвергенција грешке ка нули током итерација. Види се и да систем врло брзо постиже задовољење услова да је $|\epsilon| \leq 0.005$. Код првог зглоба максимална апсолутна вредност грешке после петнаесте итерације пада испод 0.005 rad , док код преостала два зглоба то се дешава после четврте итерације. На следећим сликама су приказане трајекторије у односу на референтну трајекторију и грешка током времена, после 150 итерација.



Слика 3.13 Жељене и стварне трајекторије(лево) и грешке(десно) за последњу итерацију

Може се видети да систем управљања постиже задовољавајуће поклапање остварене трајекторије са референтном. Примећује се да грешка код сва три зглоба има синусоиду са периодом $T \approx 6\text{s}$. Појава ове синусоиде је услед неизвесности које су придодате. И поред тога вредност грешке осцилује око нулте вредности не прелазећи задату границу.

Други случај - $\eta = \eta_2(t)$

Параметри управљања за други случај су:

$$K_p = \begin{bmatrix} 16 & 0 & 0 \\ 0 & 10 & 0 \\ 0 & 0 & 10 \end{bmatrix}, K_v = \begin{bmatrix} 8 & 0 & 0 \\ 0 & 5 & 0 \\ 0 & 0 & 7 \end{bmatrix} \quad (3.52)$$

4. Напредни алгоритам робусног управљања ИЛЦ-а целог реда ПДД2/(П)ДД2 тип у отвореној-затвореној спрези

Увидом у постојећу научну литературу која се односи на проблематику итеративног управљања учењем, могуће је уочити да многи класични ИЛЦ алгоритми имају ограничење које укључује претпоставку да матрица спрезања [СВ] има пун ранг, где В означава контролну матрицу, а С означава матрицу у простору излаза за линеарни систем, (*Arimoto S., et al. 1984*), (*Xu J.X et al, 2009*), (*Isao T. and Hunag P.H. 2017*). Да би се превазишао овај недостатак, недавно су аутори (*Lazarević et al, 2018*), сугерисали и применили ИЛЦ алгоритам целог реда у отвореној-затвореној спрези где је у отвореној грани (ПИДД2) тип, док у затвореној грани је примењен ПИД тип који успешно решава овај недостатак. Овде се излаже један робусни напредни алгоритам ИЛЦ-а (АИЛЦ) за роботски манипулатор са три степена слободе, који комбинује линеаризовани модел робота где у отвореној грани је искоришћен алгоритам (тип ПДД2) док у повратној грани је примењен (ДД2 -тип) за АИЛЦ. Стога, користећи предложену АИЛЦ шему, може се побољшати брзина конвергенције и постићи боље излазне перформансе након само неколико итерација, (*Lazarević, Mandić, Ostojić, 2020*).

Тако је у потпоглављу 4.1 презентован нелинеарни математички модел робота и генерализоване силе трења нецелобројног реда. Дизајн робусне повратне повратне спреге АИЛЦ - ПДД2 / ДД2 типа и доказ о конвергенцији предложеног ИЛЦ алгоритма представљени су у делу 4.2. На крају у потпоглављу 4.3 дате су верификације перформанси и оправданост увођења предложеног АИЛЦ на основу спроведених нумеричких симулација.

4.1 Напредно моделирање роботског система

4.1.1 Нелинеарни математички модел роботског система

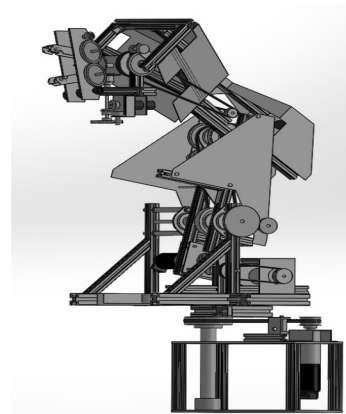
Данас роботика представља велико интердисциплинарно поље које се непрестано развија у оквиру којег се стално проналазе и реализују нове апликације. Роботи су постали саставни и неизоставан део савремене аутоматизације индустријских и неиндустријских задатака, где је потребно извршавати задатке са великом тачношћу и поновљивошћу, уз одговарајући распоред истих како би се омогућила масовна производња уз истовремено обезбеђивање квалитета производа, (*Díaz EO, 2019*), (*Kurdila AJ and Ben-Tzvi, 2020*).

Овде разматрамо *NeuroArm* роботски систем, који је саставни део Лабораторије за примењену механику Машинског факултета у Београду, и који поседује седам степени слободе, слика 4.1а. Прва три ротациона зглоба су одговорна за постављање хваталке у жељени положај (позиционирање), док следећа три зглоба чине сферни зглоб чиме се постиже потпуна оријентација роботске хваталке. Седми степен слободе поседује сама хваталка. Механичка структура *NeuroArm* робота са 7 степени слободе приказана је на слици 4.1б. У овом делу размотрићемо *NeuroArm* робот са три степена слободе са цилиндричним зглобовима (само прва три степена слободе су била активна). Овај робот има високо конфигурабилне ПИД контролере, велики интензитет обртног момента актуатора, односно Amp Maxon DC моторе са планетарним зупчаником и енкодере високе резолуције за сваки зглоб робота. Управљање се заснива на примени роботских процесора Atmega 128 са серијском комуникацијом I2C или SPI. За пројектовање напредног управљања роботом, као и реалну симулацију робота, потребан нам је прецизан динамички модел робота, (Marghita D., 2009). За ту сврху, успостављен је и представљен одговарајући модел робота *NeuroArm* користећи Родригов приступ (Човић, Лазаревић, 2009).

У циљу примене и приказа ефикасности предложеног алгоритма ИУУ даље се разматра *NeuroArm* роботска рука са прва три ротациона зглоба, тј. роботски систем који има три степена слободе, без губитка општости. Механичка структура робота *NeuroArm* се може моделовати као кинематички ланац без гранања крутих сегмената међусобно повезаних помоћу зглобова.



Слика 4.1а. Лабораторијски *NeuroArm* робот манипулатор са 7 степени слободе



Слика 4.1б. Механичка структура *NeuroArm* робота

Диференцијалне једначине кретања за дати роботски систем су дате у облику Лагранжевих једначина друге врсте као:

$$\sum_{\alpha=1}^n a_{\gamma\alpha}(q)\ddot{q}_\alpha + \sum_{\alpha=1}^n \sum_{\beta=1}^n \Gamma_{\alpha\beta,\gamma}(q)\dot{q}_\alpha\dot{q}_\beta = Q_\gamma \quad \gamma=1,2,\dots,n \quad (4.1)$$

где коефицијенти $a_{\gamma\alpha} = a_{\alpha\gamma}$ представљају коваријантне координате основног метричког тензора $[a_{\gamma\alpha}] \in R^{n \times n}$ а са $\Gamma_{\alpha\beta,\gamma}$ је означен Кристофелов симбол прве врсте. Генерализане силе Q_γ су представљене са десне стране израза (4.1) и детаљно дате у изразу (4.2) где су са $Q_\gamma^m, Q_\gamma^g, Q_\gamma^c, Q_\gamma^v, Q_\gamma^f$, $\gamma=1,2,\dots,n$ у општем случају означене генерализане управљачке силе, гравитационе силе, вискозне силе, силе у опругама, силе трења респективно:

$$Q_\gamma = Q_\gamma^m + Q_\gamma^g + Q_\gamma^v + Q_\gamma^c + Q_\gamma^f, \quad \gamma=1,2,\dots,n \quad (4.2)$$

4.2. Пројектовање контролера применом feedback линеаризације

У нашем случају, узимајући у обзир ($Q_\gamma^c, Q_\gamma^v, Q_\gamma^d, Q_\gamma^f = 0$, $\gamma=1,2,\dots,n$) динамику руке робота (4.1) можемо добити у компактном облику као:

$$a(q)\ddot{q} + (K(q, \dot{q}) - Q^g) = a(q)\ddot{q} + k(q, \dot{q}) = Q^m \quad (4.3)$$

где $a(q) = [a_{\alpha\beta}] \in R^{n \times n}$ означава матрицу инерције, $K(q, \dot{q})$ је матрица која укључује центрифугалне и Кориолисове ефекте, Q^g и $Q^m = v$ су генерализане силе од сила гравитације и управљачке силе, респективно (Човић, Лазаревић, 2009), (Kurdila AJ, Ben-Tzvi P, 2020). Увођење вектора стања као $\bar{x} = [\bar{x}_1, \bar{x}_2]^T = (q, \dot{q})^T \in R^{2n}$ може се добити (4.3) у облику простора стања:

$$\dot{\bar{x}}(t) = \begin{bmatrix} \bar{x}_2(t) \\ -a(\bar{x}_1(t))^{-1}(K(\bar{x}(t)) - Q^g) \end{bmatrix} + \begin{bmatrix} 0 \\ a(\bar{x}_1(t))^{-1} \end{bmatrix} v(t) = f(\bar{x}(t)) + g(\bar{x}(t))v(t) \quad (4.4)$$

$$y(t) = [1 \quad 0] \bar{x}(t). \quad (4.5)$$

С друге стране, предложено је много различитих шема управљања роботом, (Cvetković et al., 2018). Овде је предложено и примењена шема управљања засновану на моделу, која укључује примену технике feedback линеаризације (Khalil H, 2002), (Siciliano B. and Khatib, 2016), (Vidaković et al. 2020)

На тај начин, применом нелинеарне повратне спреге може се обезбедити поништавање нелинеарности динамике система. Као резултат, добијамо номинално линеарни систем у затвореној петљи. Линеаризација на овај начин заправо представља разлагање на унутрашњу и спољашњу петљу, где се унутрашња петља користи за линеаризацију, а спољна петља за стабилизацију система. Пре примене *feedback* линеаризације може се проверити да ли се разматрани систем може линеаризовати. Наиме, истовремено треба да буду испуњена следећа два услова (4.6-1, 4.6-2), (Slotine JJ., Li W., 1991).

а) вектори матрице $G(x) = [g(x), ad_f g(x), \dots, ad_f^{r-1} g(x)]$ су пуног ранга, где је извод Лијеве заграде $ad_f [g(x)]$ дефинисан као: (4.6а)

$$ad_f [g(x)] = \nabla g \cdot f - \nabla f \cdot g, \quad \nabla g = \frac{\partial g}{\partial x}, \nabla f = \frac{\partial f}{\partial x} \text{ и } r \text{ представља релативни ред система.}$$

б) дистрибуција $(span) \{g(x), ad_f g(x), \dots, ad_f^{r-1} g(x)\}$ је инволутивна. (4.6б)

У нашем случају, уочава се да не постоји унутрашња динамика роботског система (тј. испуњен је услов $r = n$) и добијамо линеарни систем који се састоји од n распрегнутих подсистема. Нелинеарно управљање је дато као:

$$v = (L_g L_f^{r-1} h)^{-1} (L_f^r h + u) = \dots = a(q) \cdot u + k(q, \dot{q}) \quad (4.7)$$

где L_g, L_f означавају одговарајуће Лијеве изводе, (Khalil H, 2002). У идеалном случају, узимајући у обзир да је матрица $a(q)$ ненсигуларна и (4.7), добијамо линеаризовани модел роботског манипулатора као што следи:

$$\ddot{q}(t) = u(t). \quad (4.8)$$

Линеарна динамика на горе приказани начин представља идеалан случај без утицаја неизвесности модела. Да би се узеле у обзир несигурности модела нелинеарног модела роботског система (4.1), матрице $a(q), K(q, \dot{q})$ и вектор Q^g могу се поделити на номинални и неизвесни део. С тим у вези, једначина (4.3) се може представити као:

$$a(q) \ddot{q} + (K(q, \dot{q}) - Q^g) = (a_N(q) + \Delta a) \ddot{q} + (K_N(q, \dot{q}) + \Delta K) - (Q_N^g + \Delta Q^g) = Q^m \quad (4.9)$$

Бирајући Q^m као:

$$Q^m = a_N(q)u + K_N(q, \dot{q}) - Q_N^g \quad (4.10)$$

доводи нас до:

$$\ddot{q}(t) = a^{-1}(q) a_N(q) u(t) + \eta(q, \dot{q}), \quad (4.11)$$

или у простору стања:

$$\dot{x}(t) = Ax(t) + B_a u(t) + D\eta(q, \dot{q}), \quad A = \begin{bmatrix} 0_{n \times n} & I_{n \times n} \\ 0_{n \times n} & 0_{n \times n} \end{bmatrix}, \quad B_a = \begin{bmatrix} 0_{n \times n} \\ a^{-1} a_N \end{bmatrix}, \quad D = \begin{bmatrix} 0_{n \times n} \\ I_{n \times n} \end{bmatrix} \quad (4.12)$$

$$y(t) = Cx(t) = \begin{bmatrix} I_{n \times n} & 0_{n \times n} \end{bmatrix} x(t) \quad (4.13)$$

где је $u(t)$ нови управљачки сигнал и η означава несигурност модела

$$\eta = -a^{-1}(q) (\Delta K(q, \dot{q}) - \Delta Q^g). \quad (4.14)$$

Постоји неколико начина одабира $u(t)$, укључујући неке робусне технике управљања. Један од начина за одабир $u(t)$ је као управљачки сигнал који је заснован на АИЛЦ алгоритму у отвореној- повратној спрези.

4.2.1 Робусни алгоритам АИЛЦ у отворено-затвореној спрези – PDD2/DD2 тип

У овом одељку проучавамо проблем робусног праћења поновљене путање у простору зглобова линеаризованог роботског система (4.11), $q_d(t) \in R^n \quad t \in J$, током коначног временског интервала $J = [0, T], J \subset R$ у присуству несигурности модела $\eta_i(t) \in R^n$. Овде T означава временско трајање, i -индекс итерације, $x_i \in R^n$ вектор стања, контролни улаз $u_i \in R^m$ и $y_i \in R^r$ излаз система. С друге стране, матрица [CB] има ранг у пуној колони и да бисмо превазишли ово ограничење, уведен је недавно ИЛЦ ПИДД2 / ПИД алгоритам у отвореној-повратној спрези (Lazarević PM et al.2018), који садржи други извод грешке праћења како је предложено у радовима (Fu Q, et al.2018), (Mandić, et al,2021) где је презентована супериорност ПИДЦ алгоритма који укључује и други извод грешке. Овде се предложени ИЛЦ састоји у повратној грани закона о управљању ДД2 типа и закона о директној грани ПДД2 типа (слика 4.2) за разлику од алгоритма који је дат у раду (Fu, et al.2018)., где се ИЛЦ предлаже само у директној грани, (Lazarević, Mandić, Ostojić, 2020).

Укупни улазни вектор управљања се може добити као:

$$u_{i+1}(t) = u_{f\bar{i}+1}(t) + u_{f\bar{i}+1}(t). \quad (4.15)$$

Наиме, уводимо алгоритам ИЛЦ ПДД2/ДД2 типа целог реда који је дат са:

$$u_{i+1}(t) = u_i(t) + \Gamma_1 (\ddot{e}_i(t) + R_1 \dot{e}_i(t) + R_2 e_i(t)) + \Gamma_2 (\ddot{e}_{i+1}(t) + Q_1 \dot{e}_{i+1}(t)) \quad (4.16)$$

где је $y_d(t)$ жељена излазна трајекторија, $e_i(t) = y_d(t) - y_i(t)$ и $e_{i+1}(t) = y_d(t) - y_{i+1}(t)$ су грешке праћења излазне трајекторије у i -тој итерацији, односно $i+1$ - итерацији, респективно. Такође, $\Gamma_1, \Gamma_2, Q_1, R_1, R_2 \in R^{n \times n}$ представљају позитивно дефинитне дијагоналне матрице учења. Сама примена ИУУ алгоритма захтева увођење и одговарајућих претпоставки:

A1) Жељене трајекторије $y_d(t), x_d(t)$ су непрекидно диференцијабилне на $J = [0, T]$.

A2) Понављање почетних услова из итерације у итерацију је испуњено, тј. почетно стање у i -тој итерацији је увек исто на почетку сваке итерације:

$$x_i(0) = x_d(0), \quad i = 0, 1, 2, \dots, \quad (4.17)$$

A3) Утицај промене масе роботских сегмената на матрицу $a(q)$ је занемарљив, па из тога следи да је $a^{-1}a_N \approx I$ где је I јединична матрица. Узимајући у обзир претпоставку A3, може се закључити да важи:

$$B_a = \begin{bmatrix} 0_{n \times n} \\ a^{-1}a_N \end{bmatrix} \approx B = \begin{bmatrix} 0_{n \times n} \\ I_{n \times n} \end{bmatrix} \quad (4.18)$$

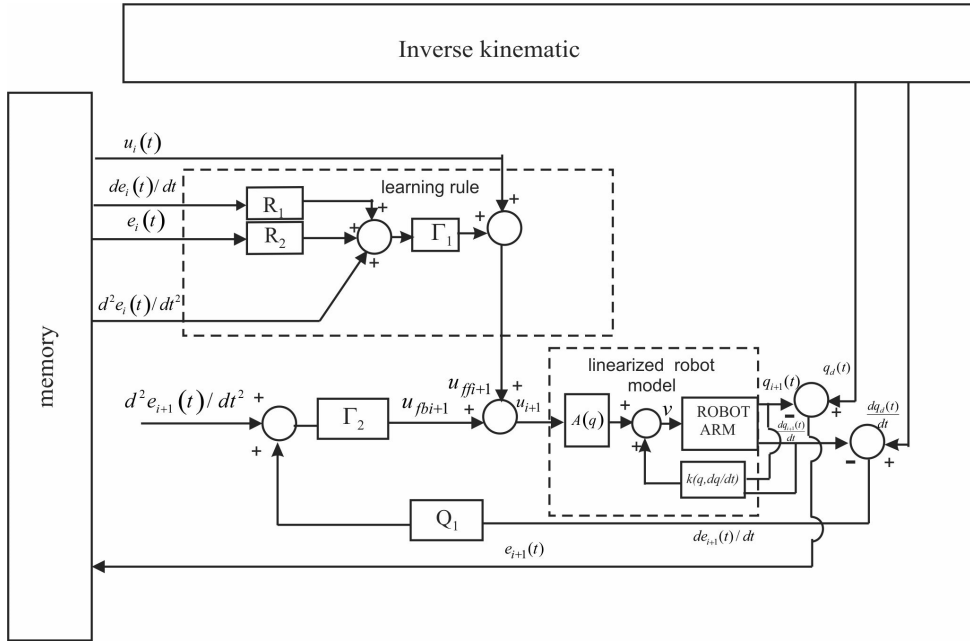
A4) Систем (4.12, 4.13) је каузалан, тј. за дат жељени излаз $y_d(t)$, постоји јединствени улазни вектор управљања $u_d(t)$, као и $x_d(t)$ који омогућавају да излаз система управо буде $y_d(t)$, тј.:

$$\dot{x}_d(t) = Ax_d(t) + Bu_d(t), \quad (4.19)$$

$$y_d(t) = Cx_d(t), \quad (4.20)$$

A5) Моделске неизвесности $\eta_i(t) \in R^n, \dot{\eta}_i(t) \in R^n$, и израз $\dot{x}_d(t)$ су униформно ограничени. У наставку уводе се позитивне константе d_η, d_{η_1}, d_x за горње границе за $\eta_i(t), \dot{\eta}_i(t)$ тј., $\forall t \in [0, T]$ и $\forall i \rightarrow \|\eta_i(t)\| \leq d_\eta, \|\dot{\eta}_i(t)\| \leq d_{\eta_1}, \|\dot{x}_d(t)\| \leq d_x$.

Довољан услов конвергенције предложеног ИЛЦ-а у отворено-затвореној спрези изведен је у временском домену и формулисан следећом теоремом Т1.



Слика 4.2. Блок дијаграм закона ИЛЦ учења PDD2/DD2 тип у отвореној-затвореној спреси за управљање робота

Доказ конвергенције:

Теорема 1. Нека је закон ИУУ (4.16) у отворено-затвореној спреси примењен на посматрани неаутономни линеаризовани систем целог реда (4.12), (4.13), нека су испуњене претпоставке A1)-A5) као и да матрица CAB има пуни ранг колоне, а матрица CB нема пуни ранг колоне. Ако постоји матрице учења Γ_1, Γ_2 такве да је следећи услов испуњен

$$\|(I + \Gamma_2 CAB)^{-1} (I - \Gamma_1 CAB)\| \leq \rho < 1 \quad (4.21)$$

и постоји матрица учења Γ_2 таква да је матрица $(I + \Gamma_2 CAB)$ инвертибилна, тада је загарантовано да се границе грешака праћења $\|x_d(t) - x_i(t)\|$, $\|y_d(t) - y_i(t)\|$, и $\|u_d(t) - u_i(t)\|$ асимптотски конвергирају у наведене границе када $i \rightarrow \infty$. Ове границе зависе од d_η, d_{η_1} .

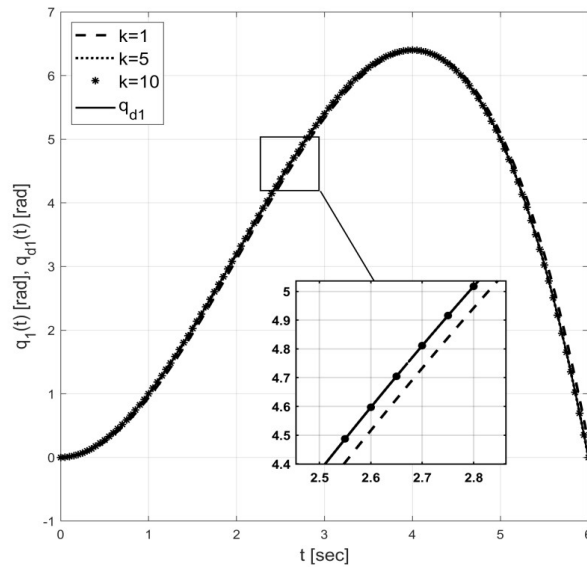
Лако се проверава да услов конвергенције датог ИЛЦ алгоритма је испуњен:

$$\|(I + \Gamma_2 CAB)^{-1} (I - \Gamma_1 CAB)\| = 0.495 < 1. \quad (4.41)$$

Појачања матрице учења R_1, R_2, Q_1 су изабране помоћу покушаја и грешке (услов конвергенције представља довољни услов), где су усвојене следеће вредности:

$$R_1 = \text{diag}(1,1,1), \quad R_2 = \text{diag}(1,1,1), \quad Q_1 = \text{diag}(20,20,5). \quad (4.42)$$

За све горе одабране параметре, резултати симулације приказани су на сл. 4.4-4.8. Сlike 4.4 и 4.6 показују да излазне трајекторије система теже жељеној трајекторији што је ближе могуће, а слике 4.5, 4.7 и 4.8 показују да ће се, уз довољан број итерација, максимална апсолутна грешка праћења конвергирати ка нули, чак и са постојањем несигурности модела.



5. Одабрани алгоритми ИУУ нецелог реда, типа D^α , PD^α

5.1 Оптимално ИУУ нецелог реда типа D^α , PD^α , $\alpha \in [0,2]$ роботског система

Овде се у овом поглављу излажу резултати који се односе на примену алгоритама управљања које садрже изводе и интеграле фракционог реда (*Kilbas et al. 2006, Lazarević, 2012; Kuzmanović et al. 2021*), уз осврт на неке најбитније резултате који се односе на ову проблематику. Даље се за један линеаризовани роботски манипулатор разматра ИЛЦ D^α , PD^α тип, $\alpha \in [0,2]$ чији су коначни (оптимални) параметри усвојени након примене погодне изабране оптимизационе методе. ИЛЦ управљање PD^α тип се затим примењује и за класу нелинеарних система са кашњењем уз пратећи доказ конвергенције истог. Ово поглавље се завршава анализом једног напредног ИЛЦ који поседује променљиво појачање и који је примењен на један нелинеарни систем са кашњењем фракционог реда.

Прво су истражене и представљене перформансе две врсте закона управљања, названих $D^\alpha ILC$ и $PD^\alpha ILC$ које су добијене применом рачуна нецелог реда и алгоритама ИУУ контролера. Представљени су неопходни услови конвергенције за предложени $PD^\alpha ILC$ за нелинеарну роботску руку. Приступ линеаризације повратном спрегом користи се заједно са класичном повратном спрегом за произвољно додељивањем полова линеаризованог система. Симулација је спроведена на линеаризованом моделу роботске руке. Предложен је *BBO* алгоритам за оптимизацију (*Simon D., 2008*), са циљем подешавања коефицијената *FOILC*. Резултати симулације показују да оптимално подешавање коефицијената *FOILC* повећава брзину конвергенције. Резултати симулације потврђују побољшање брзине конвергенције за оба типа предложених закона управљања $D^\alpha ILC$ и $PD^\alpha ILC$.

Уобичајени закони управљања итеративним учењем су пропорционалног и диференцијалног типа, (*Chen, Wen, 1999*). Закон управљања итеративним учењем диференцијалног типа садржи само извод сигнала грешке према (*Arimoto S. et al., 1984, Xu J.X, et al., 2003, 2004*):

$$u_{k+1}(t) = u_k(t) + \Gamma \frac{d}{dt} e_k(t), \quad k = 0, 1, 2, \dots \quad (5.1)$$

Слично, пропорционални закон управљања итеративним учењем се може написати као:

$$u_{k+1}(t) = u_k(t) + \Gamma e_k(t), \quad k = 0, 1, 2, \dots \quad (5.2)$$

У једначинама (5.1) и (5.2) је грешка између стварне и жељене трајекторије у k -тој итерацији и дефинише се следећом једначином:

$$e_k(t) \triangleq y_d(t) - y_k(t) \quad (5.3)$$

У претходним изразима Γ је коефицијент учења. Сигнал $y_k(t)$ је улазни управљачки сигнал у k -тој итерацији. Улаз и излаз нису познати унапред и где је параметар t представља временску променљиву, која може бити дискретна или континуална. Закон итеративног управљања учењем нецелог реда, диференцијалног (D^α), (Chen Y.Q. and K. L. Moore, 2001) и пропорционално-диференцијалног (PD^α) типа у временском домену.

$$u_{k+1}(t) = u_k(t) + k_D \frac{d^\alpha}{dt^\alpha} e_k(t) \quad (5.4)$$

$$u_{k+1}(t) = u_k(t) + k_P e_k(t) + k_D \frac{d^\alpha}{dt^\alpha} e_k(t) \quad (5.5)$$

где $\alpha = 0$ дефинише пропорционални закон управљања итеративним учењем. Слично када је $\alpha = 1$ добија се диференцијални закон управљања итеративним учењем. Изрази (5.4) и (5.5) могу се написати у s - домену као:

$$u_{k+1}(s) = u_k(s) + k_D s^\alpha E_k(s) \quad (5.6)$$

$$u_{k+1}(s) = u_k(s) + k_P E_k(s) + k_D s^\alpha E_k(s) \quad (5.7)$$

где је $E_k(s)$ Лапласова трансформација грешке $e_k(t)$. У једначинама (5.4) и (5.5), пропорционални коефицијент k_P и диференцијални коефицијент k_D су непознати константни коефицијенти учења које је потребно одредити. Утицај одабира α у интервалу $\alpha \in [0, 2]$ на конвергенцију грешке биће представљен у овом тексту.

5.1.1 Структура робота и једначине кретања

Роботске руке углавном обављају операције које се понављају, стога је значајно искористити информације о претходним операцијама у примени итеративног управљања учење чиме се побољшава одзив система и повећава тачност извођења покрета. Овде се ће искористити роботска рука са једним сегментом (Corke, et al. 1995) у циљу проучавања понашања система који је подвргнут претходно објашњеном нецелобројном итеративном управљачком закону:

$$\ddot{\theta}(t) = \frac{1}{J}(u(t) - F(t)) + \frac{1}{J}\left(\frac{1}{2}m + M\right)gl \sin \theta(t) \quad (5.8)$$

где је $\theta(t)$ угаона позиција роботске руке, управљачка величина $u(t)$ је момент управљања, $F(t)$ је момент трења, m и l су маса и дужина роботске руке респективно. M је маса врха оштрице, g је гравитационо убрзање и на крају са J је означен аксијални момент инерције зглоба. Аксијални момент инерције зглоба и момент трења су описани на следећи начин:

6. Примена ИУУ нецелог реда сингуларним системом нецелог реда који укључују и временско кашњење

У овом шестом поглављу се разматра примена ИЛЦ фракционог реда за једну ширу класу динамичких система тј. сингуларних система али сада нецелог (фракционог) реда. У уводном делу дају се рафиниране основе које се односе на ову проблематику. Затим се у наставку представља синтеза робусног управљања ИЛЦ PD^α тип сингуларних система са кашњењем фракционог реда. Изведени су довољни услови конвергенције ИЛЦ док се на крају презентују нумерички резултати који потврђују оправданост увођења једног оваквог управљања.

Наиме, извесна пажња се посвећује и проблематици управљања једном широм класом динамичких система тј. сингуларним системима нецелог реда и то применом напредних ИУУ који су такође нецелог реда. Уочено је да се сингуларни системи природно појављују у многим инжењерским дисциплинама и проблемима, где се применом рачуна нецелог реда сада могуће добити квалитетније моделе сингуларних система. Такође, овде је од интереса решавање задатка управљање истих. Са друге стране, као врста интелигентног управљања, ИУУ привукао је пажњу због једноставности и ефикасности примене алгорита ИУУ, односно његових особина да се успешно носе са проблемима неизвесности, временског кашњења, нелинеарности, као и сложенијим сингуларним системима. Сингуларни системи се добијају као резултат напредног математичког моделирања многих физичких система и процеса, и који се могу описати комбинацијом алгебарских и диференцијалних једначина. Тако, сингуларни системи се појављују у великом броју проблема практичног инжењерства, приликом формирања одговарајућег модела електричних мрежа (*Rosenbrock, 1974*) или у одређивању одговарајућих механичких модела (*Muller, 2005*) итд. На пример, електричне мреже су најчешће састављене од подсистема мрежних елемената (као што су кондензатори, отпорници, индуктивни елементи) и који су описани различитим врстама диференцијалних једначина а које се сада спрежу са Кирхофовим законима (описани алгебарским једначинама), (*Kampowski, 1996*). У механичким системима диференцијалне једначине најчешће описују динамику подсистема док су алгебарским једначинама одређена ограничења које карактеришу везе тј. ограничења која су наметнута датим везама на пример као што су зглобови (*Muller, 2005*) или у моделу ограниченог кретања роботског система тј. роботског система у контакту са околином, (*Mc-Clamroch, 1986*), (*Buzurović, 2008*).

Са друге стране, уочено је да се многи теоријски резултати који су добијени за регуларне (несингуларне) системе могу проширити на одговарајуће сингуларне

системе. У поређењу са регуларним системима, није тешко закључити да сингуларни системи могу описати и приказати ширу класу динамичких система односно описати више структурних особина физичких система него у случају регуларних система, (Debeljković, Owens 1985). У зависности од области примене, ови модели се такође називају дескриптор системи, диференцијално-алгебарским системи, системи са полустањем (*semi-state*), или уопштени системи. Посебно, дескриптивни облик укључује информације о статичким и динамичким ограничењима. Овако формирани модели сингуларних система имају извесне предности, од којих су најзначајније: задржавање битних физичких својстава у самом моделу, блиска повезаност са стварним физичким променљивама стања система, нема потребе за елиминацијом спрегнуто променљивих величина што је код нелинеарних система и практично немогуће, исписивање билансних и других једначина далеко је непосредније, матрице које фигуришу у овим моделима, по правилу су шупље, што знатно поједностављује нумеричке процедуре (Debeljković, et. al, 1996). Са друге стране, због многих специфичности сингуларних система постоји низ додатних проблема који неминовно прате њихову анализу и следствено, могућу синтезу. Наиме, код оваквих система потребно је размотрити не само стабилност истих, већ и питање постојања и јединствености решења, егзистенцију почетних услова који генеришу тзв. глатка решења и одсуство импулсног понашања у исто време (Dai,1989) за поједине сингуларне системе, приступ регуларизације сингуларних система на обичне, нормалне системе, итд. Исто тако, уочена сложена природа сингуларних система може да створи многе потешкоће у аналитичком и нумеричком решавању истих. Канонична форма сингуларних система се може приказати у облику:

$$E\dot{x}(t) = f(t, x(t), u(t)), \quad (6.1)$$

$$y(t) = g(t, x(t), u(t)) \quad (6.2)$$

или у посебном, линеарном најчешће сусретаном случају у литератури као

$$E\dot{x}(t) = Ax(t) + Bu(t), \quad (6.3)$$

$$y(t) = Cx(t) + Du(t) \quad (6.4)$$

Такође, роботски системи у решавању контактних проблема могу се моделирати као сингуларни системи, (Hui, Golenberg,1988), (Lazarević, Cvetković, Mandić 2018).

У последњих неколико деценија посебна пажња је посвећена примени рачуна нецелог реда, тј. фракционим диференцијалним једначинама и њиховим применама у различитим научним и инжењерским пољима (Kilbas et. al, 2006), (Monje et. al,2010).

Недавно по први пут примењени су алгоритми ИУУ управљања за одговарајућу класу класичних регуларних сингуларних система, (*Feng-Xian et. al,2007*), (*Lazarević,2014*).

Са друге стране, недавно су добијени су резултати у примени алгоритма ИУУ управљања нецелог реда и то за регуларне системе нецелог реда дати у временском домену, (*Lazarević, 2003*), (*Lazarević,2014*), (*Lazarević, et. al,2016*), (*Lazarević et. al,2017*)

Даље, овде ће бити презентовани нови резултати који су добијени у примени алгоритма ИУУ управљања нецелог реда а за одговарајуће класе сингуларних система нецелог реда, који се могу наћи у литератури и који све више заузимају место и у роботизици и у решавању контактних проблема.

6.1 Итеративно управљање путем учења нецелог реда за линеарне сингуларне системе нецелог реда

Овде је од интереса разматрање једног линеарног сингуларног система нецелог реда (СНРС) који је дат следећом једначином стања и једначином излаза респективно:

$$ED^\alpha x(t) = Ax(t), \quad n-1 < \alpha < n \quad (6.5)$$

$$y(t) = Cx(t), \quad (6.6)$$

При томе, дозвољени почетни услови за (6.7) дати су са:

$$x^{(k)}(t) = x_{0,k}(t), \quad k = 1, 2, \dots, n-1 \quad (6.7)$$

Овде фракциони оператор ${}_0^C D_t^\alpha = D_t^\alpha$ означава α -ти Капуто извод; односно E, A и C су матрице одговарајућих димензија.

Дефиниција 1. а) Систем СНРС-а дат са (6.5) је регуларан ако је $\det(s^\alpha E - A) \neq 0$.

б) за систем СНРС (6.5) се каже да је импулсно слободан ако је регуларан и ако је испуњено да је $\deg(\det(s^\alpha E - A)) = \text{rank} E$.

Лема 1. Тројка (E, A, α) се назива регуларном ако и само ако је $\det(s^\alpha E - A) \neq 0$ за неке $s \in \mathbb{C}$ (*Yao et. al,2013*). Ако је тројка (E, A, α) регуларна, систем СНРС (6.5) назива се регуларним, па је према томе систем СНРС решив.

Даље се посебно разматра једна класа неаутономног сингуларног линеарног система нецелог реда $\alpha \in (0,1)$ који се дат следећом једначином стања као и једначином излаза, следствено (*Sajewski,2015*):

$$ED^\alpha x(t) = Ax(t) + Bu(t), \quad 0 < \alpha < 1 \quad (6.8)$$

$$y(t) = Cx(t), \quad (6.9)$$

Овде A, B и C представљају матрице одговарајућих димензија, t означава време на одговарајућем временском интервалу рада система $J = [t_o, t_o + T], J \subset R$. Даље се уводи претпоставка да је испуњено $\det E = 0$, као и да је СНРС систем регуларан. Такође, уводи се следећа претпоставка која се односи на питање постојања, јединствености и континуитета решења у односу на почетне услове, (Yao et. al, 2013), (Ortiguiera, Coito, 2008). Са друге стране, ако постоји параметарска неизвесност код сингуларних система онда се она може приказати на следећи начин:

$$ED^\alpha x(t) = (A + \Delta A)x(t) + Bu(t), \quad 0 < \alpha < 1 \quad (6.10)$$

$$y(t) = Cx(t), \quad (6.11)$$

Уводе се и следеће претпоставке које се односе на систем (6.8), и (6.9):

A1. Жељене трајекторије $y_d(t), x_d(t)$ су непрекидно диференцијабилне на $[0, T]$.

A2. За дату жељену трајекторију излаза $y_d(t)$, постоји управљање $u_d(t)$ тако да

$$ED^\alpha x_d(t) = Ax_d(t) + Bu_d(t), \quad 0 < \alpha < 1 \quad (6.12)$$

$$y_d(t) = Cx_d(t), \quad (6.13)$$

A3. Систем СНРС је обсервабилан и управљив.

A4. За све итерације почетни услови ресетовања важе $x_k(0) = x_d(0)$, $k = 0, 1, 2, \dots$,

6.2 Итеративно управљање путем учења ПД типа нецелог реда за сингуларни систем нецелог реда са кашњењем

Даље се разматра и даје један нов резултат који се односи на ИЛЦ управљање PD^α типа у затвореној петљи за сингуларни систем са кашњењем, (Lazarević, et. al. 2017).

Неки примери сингуларних временских кашњења са нецелим редом и који су приказани у (Zhang, Wei, 2011). Размотримо следећи неаутономни нецелог реда линеарни сингуларни систем са временским кашњењем (ФОЛСТДС) описан у простору псеудо-стања и излаза:

$$ED^\alpha x(t) = A_0x(t) + A_1x(t - \tau) + Bu(t), \quad 0 < \alpha < 1 \quad (6.14)$$

$$y(t) = Cx(t), \quad (6.15)$$

са одговарајућом почетном функцијом облика:

CIP - Каталогизација у публикацији
Народна библиотека Србије, Београд

681.5:004.8

ЛАЗАРЕВИЋ, Михаило, 1964-

Итеративно управљање путем учења роботских система : класичан и модеран приступ / Михаило Лазаревић, Никола Живковић, Александар Дубоњац. - 1. изд. - Београд : Универзитет, Машински факултет, 2021 (Београд : Planeta print). - X, 280 стр. : илустр. ; 24 cm

Тираж 200. - Библиографија: стр. 271-280 и уз свако поглавље.

ISBN 978-86-6060-105-8

1. Живковић, Никола, 1992- [аутор] 2. Дубоњац, Александар Д., 1996- [аутор]
а) Итеративно управљање путем учења

COBISS.SR-ID 52593929