

Modeliranje, eksperimentalna identifikacija te dizajn estimatora stanja i regulacijskih sustava punjenja za litij-titanat baterije

Kvaternik, Karlo

Master's thesis / Diplomski rad

2023

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **University of Zagreb, Faculty of Mechanical Engineering and Naval Architecture / Sveučilište u Zagrebu, Fakultet strojarstva i brodogradnje**

Permanent link / Trajna poveznica: <https://urn.nsk.hr/urn:nbn:hr:235:803099>

Rights / Prava: [In copyright](#) / [Zaštićeno autorskim pravom](#).

Download date / Datum preuzimanja: **2024-08-08**

Repository / Repozitorij:

[Repository of Faculty of Mechanical Engineering and Naval Architecture University of Zagreb](#)



SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE

DIPLOMSKI RAD

Karlo Kvaternik

Zagreb, 2023.

SVEUČILIŠTE U ZAGREBU
FAKULTET STROJARSTVA I BRODOGRADNJE

DIPLOMSKI RAD

Mentor:

Prof. dr. sc. Danijel Pavković, dipl. ing.

Student:

Karlo Kvaternik

Zagreb, 2023.

Izjavljujem da sam ovaj rad izradio samostalno koristeći znanja stečena tijekom studija i navedenu literaturu.

Zahvaljujem svojem mentoru Prof. dr.sc. Danijelu Pavkoviću na iznimno uloženom trudu, vještima savjetima koji su poslužili kao "crna magija" te mu kao i prijatelju zahvaljujem na životnim savjetima, kolačima i razumnim rokovima. Također, htio bih zahvaliti laborantu Robertu Kamenečkom pri pomoći kod izrade makete i na korisnim savjetima.

Zahvaljujem svojim prijateljima koji su mi učinili faks lakšim te pretvorili trnoviti put u zabavni.

Zahvaljujem prijatelju Karlu koji je od početka bio velika potpora i stup razuma u teškim situacijama.

Posebno zahvaljujem svojoj obitelji, ocu Franji, majci Mandici, sestri Maji, sestri Jasni te bratu Eugenu na neizmjennoj podršci i avanturi tijekom studentskog života koji je imao svoje uspone i padove.

Na kraju zahvaljujem svojoj djevojci Matei koja me u završnici pisanja dodatno motivirala i podržavala.

Karlo Kvaternik



SVEUČILIŠTE U ZAGREBU

FAKULTET STROJARSTVA I BRODOGRADNJE

Središnje povjerenstvo za završne i diplomske ispite

Povjerenstvo za diplomske ispite studija strojarstva za smjerove:

Proizvodno inženjerstvo, inženjerstvo materijala, industrijsko inženjerstvo i menadžment, mehatronika i robotika, autonomni sustavi i računalna inteligencija



Sveučilište u Zagrebu	
Fakultet strojarstva i brodogradnje	
Datum	Prilog
Klasa: 602 - 04 / 23 - 6 / 1	
Ur.broj: 15 - 23 -	

DIPLOMSKI ZADATAK

Student: **Karlo Kvaternik**

JMBAG: **0035215265**

Naslov rada na hrvatskom jeziku: **Modeliranje, eksperimentalna identifikacija te dizajn estimatora stanja i regulacijskih sustava punjenja za litij-titanat baterije**

Naslov rada na engleskom jeziku: **Modelling, experimental identification, state estimator and charging control systems design for lithium-titanate batteries**

Opis zadatka:

Napredne sekundarne baterije temeljene na litij-titanat tehnologiji imaju niz prednosti u odnosu na druge vrste litijevih sekundarnih baterija, kao na primjer: (i) mogu se kontinuirano puniti i prazniti razmjerno velikim strujama te nisu pritom podložne stvaranju dendrita na anodi, (ii) imaju dulji životni vijek (broj ciklusa punjenja i pražnjenja) u odnosu na druge litijeve baterije i (iii) imaju visoku temperaturnu stabilnost (praktički nisu podložne efektu temperaturnog „pobjega“). S obzirom na navedeno, u radu je potrebno napraviti sljedeće:

- Dati kratki pregled i usporedbu karakteristika dostupnih baterija različitih tehnologija i ultrakondenzatora, te podrobnije opisati litij-titanat baterijsku tehnologiju.
- Opisati model nadomjesnog strujnog kruga baterije (Theveninov model) i napraviti kratki pregled drugih modela korištenih u praksi, uz kratki osvrt na termalne modele baterije pogodne za buduće primjene u optimalnim procesima punjenja baterija primjenom modelskog prediktivnog upravljanja.
- Opisati eksperimentalni postav za ispitivanje baterija i odgovarajuće identifikacijske metode za karakterizaciju nadomjesnih modela baterije: postupak određivanja krivulje nabojskog kapaciteta i karakteristike napona otvorenog kruga i eksperimentalnu identifikaciju takozvanog ARX modela u radnoj točki baterije primjenom pobude tipa pseudo-slučajnog binarnog signala (PRBS).
- Vrednovati matematički model baterije u odnosu na eksperimentalne podatke pri različitim profilima struje punjenja i pražnjenja baterije.
- Dati uvid u estimatorevarijabli stanja baterije, odabrati odgovarajuću strukturu estimatora te dizajnirati estimator stanja napunjenosti baterije.
- Projektirati konvencionalni CCCV regulator punjenja baterije u kaskadnoj strukturi upravljanja i napredni regulator punjenja temeljen na neizrazitoj (fuzzy) logici, te napraviti simulacijsku usporedbu njihovih performansi.

U radu je također potrebno navesti korištenu literaturu i eventualno dobivenu pomoć.

Zadatak zadan:

Datum predaje rada:

Predviđeni datumi obrane:

4. svibnja 2023.

6. srpnja 2023.

17. – 21. srpnja 2023.

Zadatak zadao:

Danijel Pavković

Predsjednik Povjerenstva:

Prof.dr.sc. Danijel Pavković

Ivica Garašić
Prof. dr. sc. Ivica Garašić

Sadržaj

Popis slika	III
Popis tablica	VI
Popis oznaka	VII
Sažetak	XII
Summary	XIII
1. Uvod	1
1.1. Općenito o litij-titanat baterijama	1
1.2. Usporedba baterija i ultrakondenzatora	4
1.3. Karakteristike litij-titanat baterije	6
2. Dinamički model baterije	9
2.1. RC model baterije	9
2.2. Jednostavni Theveninov model baterije	10
2.3. Theveninov model baterije	11
2.4. Postavljanje ekvivalentnog strujnog kruga	12
2.5. Termalni model baterije	14
2.5.1. Reakcijska toplina	16
2.5.2. Toplina unutarnjeg otpora	17
2.5.3. Toplina uzrokovana polarizacijskim efektom	18
2.6. Uvod u prediktivno upravljanje	19
3. Identifikacija modela baterije	22
3.1. ARX metoda	22
3.2. ARMAX metoda	23
3.3. Hammerstein-Wiener metoda	23
3.4. ARX model baterije prikladan za identifikaciju	25
3.5. Određivanje krivulje napona otvorenog kruga	33
3.5.1. Nelinearna regresija pomoću tangens funkcije	34
3.5.2. Nelinearna regresija pomoću polinomne funkcije	36
3.5.3. Nelinearna regresija pomoću kubičnog spline-a	36
3.6. Određivanje krivulje nabojskog kapaciteta	40

3.7. Validacija matematičkog modela baterije	41
4. Estimator stanja	46
4.1. Kalmanov filter	46
4.2. Prošireni Kalmanov filter	49
4.3. Estimator s pomičnim horizontom	50
4.4. Estimator stanja napunjenosti baterije na temelju proširenog Kalmanovog filtra	52
5. Dizajn sustava upravljanja	57
5.1. Općenito o sustavima PI upravljanja	57
5.2. Konvencionalni sustav upravljanja punjenja – referentni slučaj	58
5.3. Upravljanje temeljeno na neizrazitoj logici	63
5.3.1. Fazifikacija	63
5.3.2. Neizrazito odlučivanje	64
5.3.3. Defazifikacija	64
5.4. Dizajn neizrazitog regulatora	65
5.5. Usporedba između neizrazite logike i konvencionalnog sustava upravljanja punjenjem baterije	68
6. Zaključak	72
Bibliografija	XIV
Prilog	XVI

Popis slika

1.	Karakteristike baterije (1: Loše karakteristike, 2: Odlične karakteristike) .	3
2.	Shematski prikaz ultrakondenzatora	4
3.	Strujni krug ultrakondenzatora	5
4.	Pojednostavljeni Strujni krug ultrakondenzatora	5
5.	Karakteristike baterije te ultrakondenzatora	6
6.	RC model baterije	9
7.	Jednostavni Theveninov baterije	10
8.	Theveninov model	11
9.	Theveninov model sa 2 RC strujna kruga	11
10.	RC strujni krug	13
11.	Shema litij-ionske ćelije	15
12.	Princip rada prediktivnog upravljanja	19
13.	Napon baterije uzrokovan PRBS-om te aproksimacija napona otvorenog kruga	26
14.	Postava za testiranje baterije	27
15.	Usporedni eksperimentalni tragovi tijekom punjenja i pražnjenja s prosječnom (DC) strujom od 24 A sa superponiranim perturbacijama PRBS signala od ± 4 A.	28
16.	Unutarnji otpor baterije u odnosu sa stanjem napunjenosti - punjenje . . .	29
17.	Unutarnji otpor baterije u odnosu sa stanjem napunjenosti - pražnjenje . .	30
18.	Polarizacijski otpor baterije u odnosu sa stanjem napunjenosti - punjenje .	30
19.	Polarizacijski otpor baterije u odnosu sa stanjem napunjenosti - pražnjenje	31

20. Polarizacijska vremenska konstanta baterije u odnosu sa stanjem napunjenosti - punjenje	31
21. Polarizacijska vremenska konstanta baterije u odnosu sa stanjem napunjenosti - pražnjenje	32
22. Mjerene točke odvojene po segmentima	37
23. Krivulje napona otvorenog kruga U_{oc}	39
24. Krivulje pražnjenja konstantne struje baterije (a) i ovisnost kapaciteta baterije u odnosu na struju pražnjenja (b)	40
25. Blok dijagram ekvivalentnog strujnog kruga baterije	41
26. Usporedba simulacije i eksperimentalnog rezultata za napon baterije tijekom punjenja sa strujom baterije od 6 A	42
27. Usporedba simulacije i eksperimentalnog rezultata za napon baterije tijekom pražnjenja sa strujom baterije od 6 A	42
28. Usporedba simulacije i eksperimentalnog rezultata za napon baterije tijekom punjenja sa strujom baterije od 12 A	43
29. Usporedba simulacije i eksperimentalnog rezultata za napon baterije tijekom pražnjenja sa strujom baterije od 12 A	43
30. Usporedba simulacije i eksperimentalnog rezultata za napon baterije tijekom punjenja sa strujom baterije od 18 A	44
31. Usporedba simulacije i eksperimentalnog rezultata za napon baterije tijekom pražnjenja sa strujom baterije od 18 A	44
32. Usporedba simulacije i eksperimentalnog rezultata za napon baterije tijekom punjenja sa strujom baterije od 24 A	45
33. Usporedba simulacije i eksperimentalnog rezultata za napon baterije tijekom pražnjenja sa strujom baterije od 24 A	45
34. Kalman filter kao estimator stanja	47
35. Estimator pomičnog horizonta	51
36. Blok dijagram estimatora stanja	55
37. Usporedba estimirane i simulacijske vrijednosti stanja napunjenosti tokom punjenja konstantnom strujom	56
38. Uvećani dio početnog dijela	56
39. Kaskadna regulacija punjenja baterije	59

40.	Podređeni regulacijski krug struje	59
41.	Nadređeni regulacijski krug napona	60
42.	Blok dijagram neizrazitog regulatora kao punjač baterije	65
43.	Funkcije pripadnosti neizrazitog regulatora s obzirom na signal pogreške upravljanja SoC-a	66
44.	Funkcije pripadnosti neizrazitog regulatora s obzirom na signal napona baterije	66
45.	Vrijednosti izlaznog pravila solitona	67
46.	Struja punjenja; PI vs Neizrazitog regulatora	68
47.	Napon pretvarača; PI vs Neizraziti regulator	69
48.	Napon baterije; PI vs Neizraziti regulator	70
49.	Stanje napunjenosti baterije; PI vs Neizraziti regulator	70
50.	Uvećani početni dio prethodne slike	71

Popis tablica

1.	Elektrokemijska i mehanička svojstva	7
2.	Karakteristike baterijskih ćelija koristeći grafit i LTO kao anodu	7
3.	Mjerene vrijednosti napona otvorenog kruga	34
4.	Vrijednost polinoma	36

Popis oznaka

Oznaka	Jedinica	Opis
A	-	Matrica sustava vremenski kontinuiranog modela u prostoru stanja
B	-	Matrica ulaza vremenski kontinuiranog modela u prostoru stanja
C	-	Matrica izlaza vremenski kontinuiranog modela u prostoru stanja
C_b	F	Kapacitet <i>Bulk</i> kondenzatora
C_c	F	Kapacitet „površinskog“ kondenzatora
C_p	F	Parazitska kapacitivnost
D	-	Matrica prijenosa direktne putanje vremenski kontinuiranog modela u prostoru stanja
e(k)	-	Vektor šuma mjerenja
F	Cmol^{-1}	Faradayeva konstanta
F	-	Matrica sustava vremenski diskretnog modela u prostoru stanja
G	-	Ulazna matrica vremenski diskretnog modela u prostoru stanja
H	-	Izlazna matrica vremenski diskretnog modela u prostoru stanja
I	-	Jedinična matrica
i_b	A	Struja baterije
i_r	A	Struja otpornika
i_c	A	Struja kondenzatora

Oznaka	Jedinica	Opis
j	A/cm ²	Gustoća reakcijske struje u elektrodi
$\mathbf{K}(k)$	-	Matrica pojačanja Kalmanovog filtra
L	cm	Udaljenost polova
n_c	-	Broj elektrona
$\mathbf{P}(k k-1)$	-	A priori procjena matrice kovarijanci \mathbf{P}
$\mathbf{P}(k k)$	-	A posteriori procjena matrice kovarijanci \mathbf{P}
\mathbf{Q}	-	Matrica kovarijanci stohastičkih perturbacija u varijablama stanja
S_e	cm ²	Aktivna površina elektrode
T_{bat}	K	Temperatura baterije
Q_b	Ah	Trenutni naboj u bateriji
Q_G	W	Ukupna disipirana toplina
Q_{irr}	W	Ireverzibilna disipirana toplina
Q_o	W	Toplina uzrokovana unutarnjim otporom
Q_p	W	Toplina uzrokovana polarizacijskim efektom
Q_r	W	Reakcijska toplina
Q_{rev}	W	Reverzibilna disipirana toplina
\mathbf{Q}	-	Matrica kovarijanci stohastičkih perturbacija u varijablama stanja

Oznaka	Jedinica	Opis
\mathbf{R}	-	Matrica kovarijanci šuma mjerenja
R_b	Ω	Unutarnji otpor baterije
R_c	Ω	Otpor dielektrika
R_e	Ω	Otpor vodova
R_p	Ω	Polarizacijski otpor
R_{p1}	Ω	Polarizacijski otpor sporije dinamika
R_p	Ω	Polarizacijski otpor brže dinamike
SoC, ξ	%	Stanje napunjenosti baterije
T	s	Vrijeme uzorkovanja estimatora
\mathbf{u}	Ah	Vektor ulaza
u_{bat}, U_{bat}	V	Napon baterije
U_b	V	Napon <i>bulk</i> napona
U_c	V	Napon „površinskog“ kondenzatora
u_p, U_p	V	Napon polarizacije
u_{oc}, U_{oc}	V	Napon otvorenog kruga
$\boldsymbol{\nu}$	-	Vektor stohastičkih perturbacija varijabli stanja \mathbf{x}
τ_p	s	Vremenska konstanta polarizacijskog efekta

Oznaka	Jedinica	Opis
\mathbf{x}	-	Vektor stanja
$\hat{\mathbf{x}}(k k-1)$	-	Apriori procjena varijabli stanja
$\hat{\mathbf{x}}(k k)$	-	Aposteriori procjena varijabli stanja
$\mathbf{y}(k)$	-	Vektor izlaznih varijabli stanja
$\hat{\epsilon}(k k-1)$	-	Procjena predikcijske pogreške Kalmanovog filtra
ΔS	$\text{Jmol}^{-1}\text{K}^{-1}$	Promjena entropije elektrodnog materijala
α_a, α_c	-	Anodni i katodni prijenosni koeficijenti
δ	m	Debljina termalnog površinskog sloja
δ_n	cm	Debljina negativne elektrode
δ_p	cm	Debljina pozitivne elektrode
δ_{sep}	cm	Debljina separatora
ϵ_e	-	Poroznost elektrode
η	V	Aktivacijski prenapon
κ^{eff}	Scm^{-1}	Efektivna ionska provodljivost elektrolitske faze
κ_D^{eff}	Acm^{-1}	Efektivna ionska difuzijska provodljivost elektrolitske faze
λ_{air}	$\text{W m}^{-1} \text{K}^{-1}$	Toplinska provodljivost zraka
σ^{eff}	S cm^{-1}	Efektivna elektronska vodljivost čvrste faze

Oznaka	Jedinica	Opis
ϕ_e	V	Električni potencijal elektrolitske faze
ϕ_s	V	Električni potencijal čvrste faze

Sažetak

Među mnogim punjivim litijevim baterijama, litij-titanat ćelije karakteriziraju najviša razina toplinske stabilnosti i radne sigurnosti, što ih čini posebno prikladnima za visoko zahtjevne primjene. Ovaj rad predstavlja identifikaciju, rezultate eksperimentalne karakterizacije ćelije litij-titanat baterije te dizajn CCCV sustava upravljanja punjenja baterije korištenjem strukture kaskadnog sustava upravljanja u kojem se stanje napunjenosti baterije kontrolira na takav način da se postigne potpuno napunjeno stanje u minimalnom vremenu, poštujući ograničenja napona baterije i struje punjenja. CCCV sustav je korišten kao usporedba za punjač temeljen na neizrastoj logici koji imitira pulsno punjenje. Identifikacija modela baterijskih ćelija provedena je u tri faze: (i) snimanje napona otvorenog kruga u odnosu na karakteristike stanja napunjenosti pomoću punjenja slabom strujom isprekidanog s intervalima rekuperacije napona baterije u cijelom rasponu baterije napona, (ii) određivanje efektivnog kapaciteta punjenja baterije i (iii) identifikaciju serijskog otpora baterijskih ćelija i polarizacijskog otpora i vremenske konstante kašnjenja za različite vrijednosti struje punjenja korištenjem rekurzivne procjene parametra na temelju testnog signala pseudoslučajnog binarnog niza. Povratna informacija o stanju napunjenosti osigurava se korištenjem proširenog Kalmanovog filtra koji uključuje realistični nelinearni Theveninov model ekvivalentnog kruga baterijske ćelije. Učinkovitost predloženog estimatora i neizrastog punjača provjerena je pomoću simulacija.

Ključne riječi: Litij-titanatna baterijska ćelija; Model ekvivalentnog kruga; Eksperimentalna identifikacija; Procjena stanja; Prošireni Kalmanov filtar; CCCV metoda punjenja, Punjenje temeljeno na neizrastom upravljanju.

Summary

Among the many rechargeable lithium batteries, lithium-titanate cells are characterized by highest thermal stability and operational safety levels, which makes them particularly well suited for highly-demanding applications. This paper presents the results of experimental characterization of a lithium-titanate battery cell and design of a CCCV battery charging control system using a cascade control system arrangement wherein the battery state-of-charge is controlled in such a way so as to reach the fully charged state in a minimum time, while honoring the battery cell voltage and charging current constraints. CCCV system is used as a comparison for fuzzy charging control system which imitates pulse charging. Battery cell model identification has been carried out in three stages: (i) recording of the open-circuit voltage vs. state-of-charge characteristic by means of low-current charging interspersed with battery voltage recuperation intervals over the full-range of battery voltages, (ii) determination of the effective battery charge capacity, and (iii) identification of battery cell series resistance and polarization resistance and lag time constant for different charging current values using recursive parameter estimation based on pseudo-random binary sequence test signal. The state-of-charge feedback is provided by using an extended Kalman filter which incorporates the realistic nonlinear Thevenin model of the battery cell equivalent circuit. The effectiveness of the proposed estimator and fuzzy charger is verified by means of simulations.

Keywords: Lithium-Titanate Battery Cell; Equivalent Circuit Model; Experimental Identification; State-of-Charge Estimation; Extended Kalman Filter; CCCV Charging; Fuzzy Charging Control.

1. Uvod

1.1. Općenito o litij-titanat baterijama

Litijeve baterije predstavljaju najsuvremeniju tehnologiju elektrokemijskih baterija s povoljnim kalendarskim vijekom trajanja do 15 godina i specifičnim troškovima od oko 145 EUR/kWh pohranjene električne energije za najnapredniju tehnologiju baterija litij-titanat [1]. Zbog kapaciteta baterije, ona još uvijek predstavlja rješenje povoljnije od alternativnih izvora energije kao što je vodik, posebno kada se uzme u obzir povratna učinkovitost pretvorbe energije [1]. Anodni materijal litij-titanat oksid predložen je prije desetak godina kao dobra zamjena za uobičajeno korištene anode na bazi ugljika u sekundarnim baterijama na bazi litija zbog praktične odsutnosti naprezanja anodnog materijala tijekom pražnjenja visokih struja [2]. Zapravo, litij-titanat oksid se također može koristiti kao anodni materijal u drugim, alternativnim baterijskim tehnologijama kao što su one temeljene na natrij-ionskoj tehnologiji, čija bi proizvodnja mogla biti jeftinija zbog prirodnog obilja natrija kao sirovine [3].

Štoviše, napredne baterije na bazi litija, kao što su litij-željezo-fosfat (LiFePO_4) i litij-titanat (LTO) su trenutno među najsigurnijim dostupnim tehnologijama za pohranu energije u baterijama, koje karakterizira iznimna toplinska stabilnost i životni vijek (trajnost) [4]. Prethodno navedene tehnologije baterija također su komplementarne u smislu energije i gustoće snage (pri čemu su LiFePO_4 baterije prikladnije za dugotrajno pražnjenje niske snage, a LTO ima veću brzinu punjenja i pražnjenja). Stoga su ove baterije razmatrane unutar hibridnog baterijskog skladištenja energije, što je rezultiralo 10-20 % nižim troškovima u usporedbi s baterijskim paketima koji se temelje isključivo na LTO i LiFePO_4 [4].

U tu svrhu, referenca [5] raspravlja o ključnim prednostima LTO ćelija, s dodatnim prednostima u smislu poboljšanih toplinskih svojstava materijala u usporedbi s drugim litij-ionskim kemijama što dovodi do poboljšanih sigurnosnih značajki, kao što je značajno smanjenje stvaranja dendrita i litija oplata elektroda, čak i pod teškim radnim uvjetima, kao što su visoke struje (oko 25C za kontinuirano pražnjenje i punjenje), zajedno s iznimnom izdržljivošću (20000 ciklusa punjenja/pražnjenja s 1C strujom). Studija u [5] zaključuje da LTO baterijske ćelije nude iznimne mogućnosti brzine punjenja/pražnjenja, iako sa smanjenom gustoćom energije u usporedbi s klasičnim litij-ionskim ćelijama s grafitnim anodama. Zbog svojih iznimnih cikličnih karakteristika i izdržljivosti, čini se da su LTO ćelije prikladne za hibridne primjene energije u pogonskim sklopovima elektrificiranih vozila.

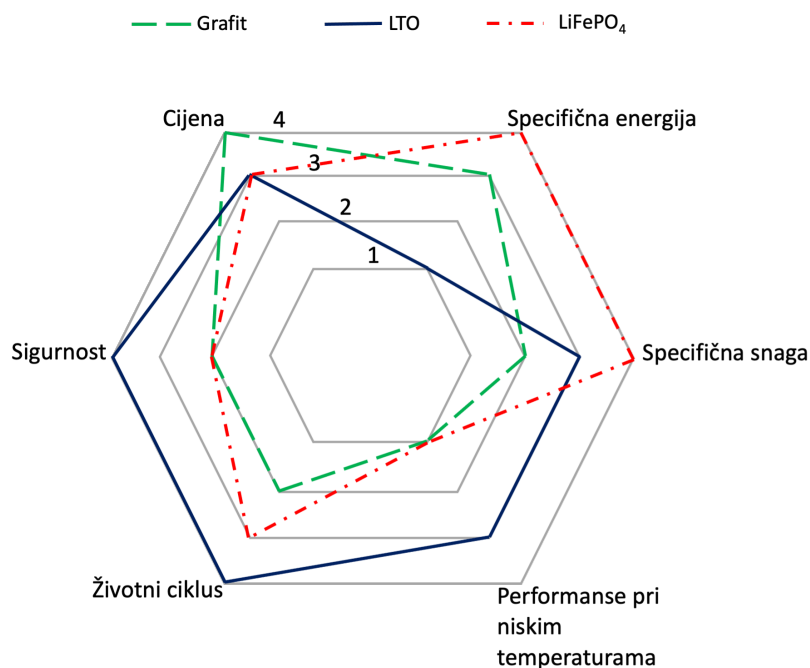
Kako bi se olakšao povećani prodor tehnologija litijevih baterija u heterogone domene, najprije je potrebno provesti njihovo modeliranje i eksperimentalnu identifikaciju, pri čemu se dobiveni model baterije može koristiti u brojne svrhe, kao što su: (i) termički nadzor temeljen na smanjen skup mjerenja [6], (ii) stanje napunjenosti (SoC) i praćenje stanja energije [7], obično unutar prikladnog okvira procjene [8], (iii) dizajn sustava kontrole punjenja baterije [9] i dizajn sustava upravljanja baterijom [10], i (iv) procjena zdravstvenog stanja baterije (SoH) [4] i preostalog vijeka trajanja (RUL) [11]. Takvi napori na modeliranju baterija obično su usmjereni na razvoj modela ekvivalentnog električnog kruga (tzv. Theveninov model) [12], koji također može uključivati toplinske učinke [6] prvenstveno povezane s temperaturnim varijacijama parametara Theveninovog modela [13]. Alternativno, može se upotrijebiti napredniji model koji uključuje elektrokemijske fenomene uz toplinske učinke [14], koji se zatim mogu upotrijebiti za analizu upravljanja toplinom baterije temeljenu na metodi konačnih elemenata (FEM) [15].

Jedan od ključnih zahtjeva je pravovremeno punjenje baterija električnih vozila, kao što je održavanje optimalnih profila stanja napunjenosti baterije (SoC) [6], što naglašava potrebu za učinkovitim punjenjem baterije korištenjem odgovarajućih strategija kontrole punjenja. U primjenama punjenja baterija, takozvana tehnika punjenja konstantnom strujom pri konstantnom naponu (engl. Constant-Current Constant-Voltage ili CCCV) [7] obično se preferira kada se baterija puni iz ispražnjenog stanja [8], dok su pulsno punjenje (vidi npr. [6]) i sporo (*trickle*) punjenje pristupi punjenju [10] su obično rezervirani za održavanje napunjenosti baterije nakon što je potpuno napunjena (tzv. plutajuće punjenje). Glavna prednost CCCV pristupa je da su struja punjenja i napon na stezaljkama baterije učinkovito ograničeni pomoću odgovarajućih regulatora struje i napona na bateriji, čime se smanjuju toplinska opterećenja baterije i sprječavaju prenaponi baterije. Međutim, baterije obično pokazuju značajno nelinearno ponašanje napona na stezaljkama tijekom punjenja [12], što može biti izazovno sa stajališta preciznog praćenja SoC baterije. To je ključno za sprječavanje prekomjernog punjenja baterije i povezanog starenja ćelija i Stanje ispravnosti (SoH)[13]. Iz tog razloga, stanje napunjenosti baterije treba nadzirati tijekom cijelog procesa punjenja pomoću odgovarajućeg estimatora, kao što je prošireni Kalmanov filter [12], a koji je uspješno primijenjen u [14] za proširenje konvencionalnog CCCV sustava punjenja, što je u konačnici rezultiralo s oko 25%-tnim ubrzanjem procesa punjenja.

Daljnja poboljšanja trajanja procesa punjenja i mogućnosti praćenja SoC-a vjerojatno su moguća uz pomoć naprednih strategija upravljanja. Na primjer, komparativna studija predstavljena u [15] je istaknula da korištenje pristupa baziranog na neizravnoj logici u sustavu upravljanja [14] može dati oko 13% poboljšanja u trajanju punjenja baterije, dok modelsko prediktivno upravljanje (MPC) s estimatorom SoC baterije temeljenom na principu pomičnog horizonta [17], mogu rezultirati relativnim ubrzanjima procesa pu-

njenja od 17% odnosno 30%.

Imajući na umu ove aspekte korištenja baterije, ovaj rad predstavlja rezultate eksperimentalne karakterizacije najsvremenije LTO baterijske ćelije, koja se koristi za modeliranje i validaciju Theveninova modela sa grupiranim parametrima usmjerenim na kontrolu baterijske ćelije. Model se kasnije koristi za izradu EKF-a temeljenog estimatora stanja napunjenosti baterije, što će se u radu i objasniti. Sa gotovim modelima, kasnije ćemo u radu razviti pulsni punjač zasnovan na neizrazitoj logici te CCCV punjač s kojima ćemo usporediti njihove karakteristike.



Slika 1: Karakteristike baterije (1: Loše karakteristike, 2: Odlične karakteristike)

Ako pogledamo sliku 1, vidimo usporedbu tri tipa elektrokemija baterije; Litij-ionsku sa grafitnom anodom, Litij-titanat te Litij-željezno fosfatna baterija. Svaki tip baterije ima neke performanse bolje od drugih, no glavna stvar pri ispravnom odabiru je njena svrha korištenja. Navedene baterije su konkurenti u svojem području. Također, sistem ocjenjivanja baterija je napravljen preko šesterokuta u kojem su navedene ključne karakteristike sa bodovim sustavom. Ako uzememo da svaka stavka ima jednaku vrijednost (da bude najbolji izbor u svakom području) tu prevladava litij-titanat baterija, koju dosta dobro prati litij-željezno fosfatna baterija.

Ovakav tip baterija je predviđen za:

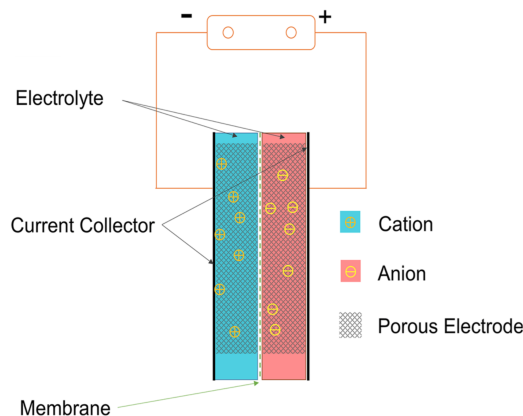
- **Hibridna električna vozila** - za *mild* hibride, no moguće i za full hibridna, no sa ograničenom brzinom

- **Električni busevi** - u potpuno električni busevi sa rutama na kojima se nalaze brzi punjači
- **Hibridni sustavi za pohranu energije** - baterijski paketi koji služe za pokrivanje vršnih (*peak*) snaga koji funkcionira sa glavnim sustavom baterije

Zanimljivo je da ovakav tip baterije može poslužiti kao sekundarna baterija koja obavlja dosta sličnu stvar kao i već poznat ultrakondenzator. Nadalje, u sljedećem odjeljku ćemo provesti analizu takvih sekundarnih baterija sa ultrakondenzatorima te objasniti njihovu kombinaciju.

1.2. Usporedba baterija i ultrakondenzatora

Ultrakondenzator je dvoslojni elektrokemijski kondenzator koji može pohraniti do 1000 puta više energije nego običan kondenzator. Oni su, po matematičkom opisu, vrlo slični baterijama (iako se zasnivaju na pohrani pomoću separacije naboja, a ne elektrokemijskim reakcijama), no same vrijednosti parametara se dosta razlikuju. Glavna i bitna razlika između njih je da ultrakondenzator ne može funkcionirati kao primarni sustav za pohranu energije jer nema dostatan kapacitet (tipično manje od 20 % od tipičnog kapaciteta baterije).



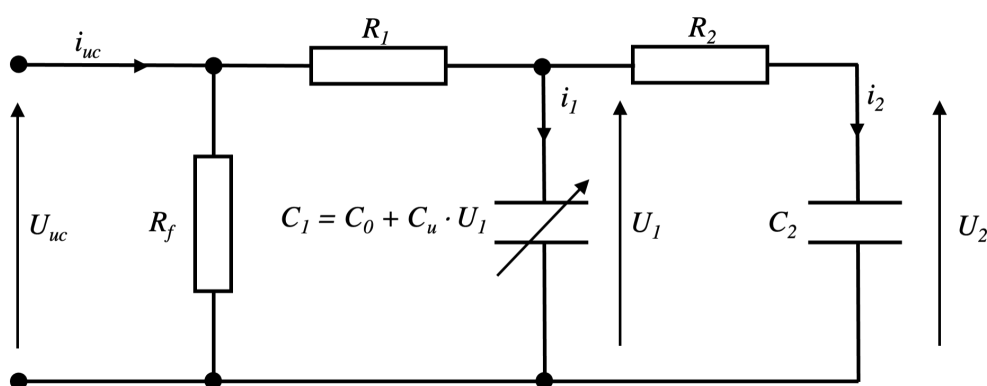
Slika 2: Shematski prikaz ultrakondenzatora

Druga glavna razlika je da su baterije najefikasnije kada se koriste za opskrbu niskih, relativno stalnih razina snage, dok se ultrakondenzatori mogu bez gotovo ikakvih nuspojava koristiti za impulsna opterećenja, točnije pokrivanje vršnih nepredvidivih snaga. Upravo tim dobivamo ugađeniju dostavu energiju prema ostalim elektrokompnentama. Kako je već prije spomenuto, pohrana energije u baterijama temelji se na elektrokemijskim reakcijama. Energija sadržana u kemijskim vezama aktivnog materijala vraća se u električnu energiju kroz niz reakcija oksidacije / redukcije (redoks reakcije).

Prilikom ponovnog punjenja, proces je obrnut. Ovaj tip reakcije uključuje prijenos elektrona s jednog materijala na drugi kroz električni krug. Mogu postojati elektrokemijske reakcije, no one su sekundarne. Primarno je to elektrostatička separacija naboja.

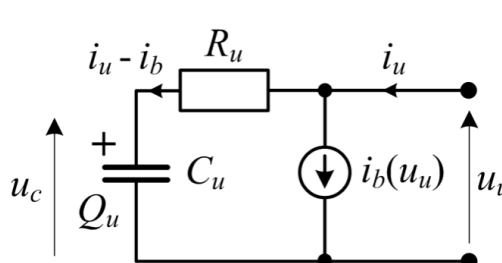
Kao što je vidljivo na slici 2, sastoji se od 3 glavnih komponentata:

- **Anoda ili negativno nabijena elektroda** - oksidira tj. otpušta elektrone kroz strujni krug.
- **Katoda ili pozitivno nabijena elektroda** - reducira tj. prihvaća elektrone.
- **Elektrolit** - medij za prijenos naboja, iona unutar ćelije između anode i katode.



Slika 3: Strujni krug ultrakondenzatora

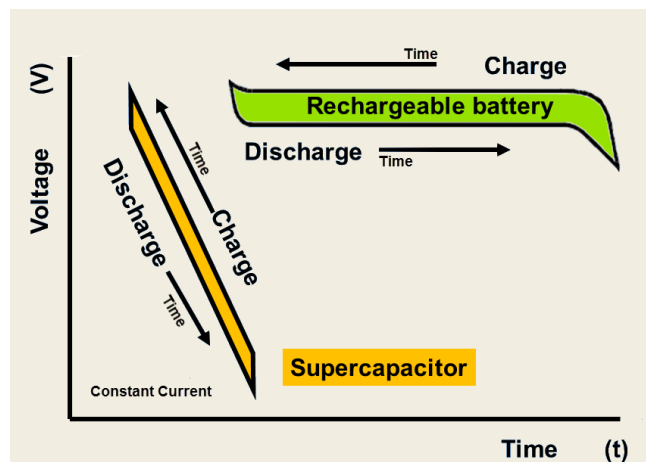
Sa slike 3, Glavni kondenzator C_1 ovisi o naponu U_1 . Sastoji se od konstante vrijednosti C_0 i konstantog parametra koji je ovisan o naponu C_u ($\frac{F}{V}$). $R_1 C_1$ grana određuje trenutno stanje ultrakondenzatora tijekom brze promjene punjenja ili pražnjenja. $R_2 C_2$ je grana koja opisuje sporu dinamiku. Opisuje distribuciju unutarnje energije pri kraju punjenja. Naravno, ovakav opis je detaljan te ne stvara preveliku razliku u rezultatima. Stoga, opis strujnog kruga izgleda:



Slika 4: Pojednostavljeni Strujni krug ultrakondenzatora

Kasnije u radu ćemo vidjeti da pojednostavljeni strujni krug dosta slično izgleda kao i Theveninov model. Slika 5 odlično prikazuje razliku koju smo spomenuli u početku;

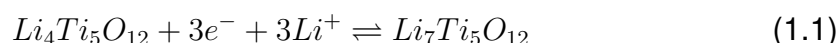
Daleko manji kapacitet posjeduje, no može pokriti velike struje dok baterija posjeduje veći kapacitet, no predviđena je za stabilnije, manje iznose struja.



Slika 5: Karakteristike baterije te ultrakondenzatora

1.3. Karakteristike litij-titanat baterije

Za katodu ćelije litij-ionske baterije višestruki materijali poput oksida prijelaznih metala (litij kobalt oksid - LCO, litij mangan oksid - LMO, nikal kobalt aluminijev oksid - NCA, nikal mangan kobalt oksid - NMC) ili fosfati (litij željezo fosfat - LFP) etablirali su se zbog svojih visokih redoks potencijala u odnosu na Li/Li+. Svaki od ovih materijala nudi pojedinačne prednosti u pogledu gustoće energije, snage, sigurnosti, životnog vijeka i troškova. Stoga se svi oni obično koriste u najsuvremenijim Li-ion baterijskim ćelijama [12,13]. Nasuprot tome, samo se grafit koristi kao anodni materijal u većini komercijalnih aplikacija zbog njegovog niskog redoks potencijala u odnosu na Li/Li+, što omogućuje visoke napone ćelija. Nadalje, grafit ima visok specifični kapacitet što dovodi do visoke gustoće energije. Novi materijali kao što su kompoziti silicij-ugljik, litij-silicijeve legure ili litijevi metali obećavajući su kandidati za poboljšanje performansi, ali su trenutno još uvijek predmet intenzivnih istraživačkih aktivnosti [14-17]. Zamjena grafita LTO-om dovodi do promjene električnih, toplinskih svojstava i svojstava starenja baterije. Zbog niskog teorijskog specifičnog kapaciteta spinel strukturiranog LTO-a i visokog radnog potencijala od 1,55 V tijekom litiranja/delitiranja (što rezultira nižim naponom pune ćelije), gustoća energije baterijske ćelije je smanjena. Teoretski specifični kapacitet od 175 mAh/g može se dobiti iz jednadžbe koja opisuje proces umetanja/vađenja litija :



S druge strane, veći radni potencijal i toplinska svojstva materijala dovode do povećane sigurnosti [18]: Rizik od stvaranja sloja sučelja čvrstog elektrolita (SEI) na

površini anode zbog redukcije elektrolita je eliminiran [15]. Nadalje, stvaranje dendritičkog litija je smanjeno i ne dolazi do presvlačenja litijem, čak ni kada se radi na temperaturama ispod 0°C. Nano-strukturiranje, dopiranje i površinsko premazivanje metode su za povećanje niske električne vodljivosti i loših svojstava difuzije litija i na taj način omogućavaju razmjerno visoke gustoće struje na elektrodama. Glavna prednost LTO-a proizlazi iz njegove karakteristike "nultog naprezanja": umetanje/ekstrakcija litija odvija se u vrlo reverzibilnoj reakciji koja se odvija bez značajne promjene volumena. Zbog toga se unutar materijala tijekom ciklusa javlja minimalno mehaničko naprezanje, što dovodi do izvrsnog životnog vijeka čak i kod ciklusa pri dubini pražnjenja (DOD) od 100% [19,20]. Ovisno o uvjetima ispitivanja (npr. temperatura okoline, struja ciklusa, DOD), proizvođači ćelija navode životni vijek od 15 000 – 60 000 punih ciklusa za ćelije s LTO anodama. Osim svojih elektrokemijskih svojstava, LTO je netoksičan, ekološki benigni materijal koji se može proizvesti uz niske troškove [15].

Tablica 1 prikazuje različita elektrokemijska svojstva dvaju anodnih materijala grafit i LTO.

Tablica 1: Elektrokemijska i mehanička svojstva

		Grafit	LTO
Teorijski specifični kapacitet	mAh/g	372	275
Razlika potencijala Li/Li ⁺	V	0.1	1.55
Električna provodljivost	S/cm	10 ⁻⁴	10 ⁻¹³ -10 ⁻⁸
Koeficijent litijske difuzije	cm/s	10 ⁻¹¹ -10 ⁻⁷	10 ⁻¹⁶ -10 ⁻⁸
Volumenska promjena tokom litizacije/delitizacije	%	10	0.2

Tablica 2: Karakteristike baterijskih ćelija koristeći grafit i LTO kao anodu

		Grafit	LTO
Nominalni napon ćelije	V	3.2 - 3.7	1.9 - 2.5
Specifična gustoća energije	Wh/kg	130 - 260	45 - 100
Specifični volumen energije	Wh/l	300 - 700	80 - 250
Životni ciklus		500 - 4000	15,000 - 60,000
Temperaturni raspon(Punjenje / Pražnjenje)	°C	0 - 45 / -20 - 60	-30 - 60 / -30 - 60

Svojstva pune ćelije određena su ne samo materijalima anode i katode, već i ostalim komponentama ćelije (elektrolit, separator, kolektor struje, kućište). Tablica 2 prikazuje raspon svojstava ćelija na bazi grafit i LTO. Treba napomenuti da se podaci u tablici odnose na standardne masovno proizvedene ćelije Li-ion baterije. U visoko specijaliziranim primjenama (npr. zrakoplovstvo), neka svojstva ćelija mogu se razlikovati od prikazanog raspona. Općenito, svojstva se razlikuju ovisno o korištenim materijalima i

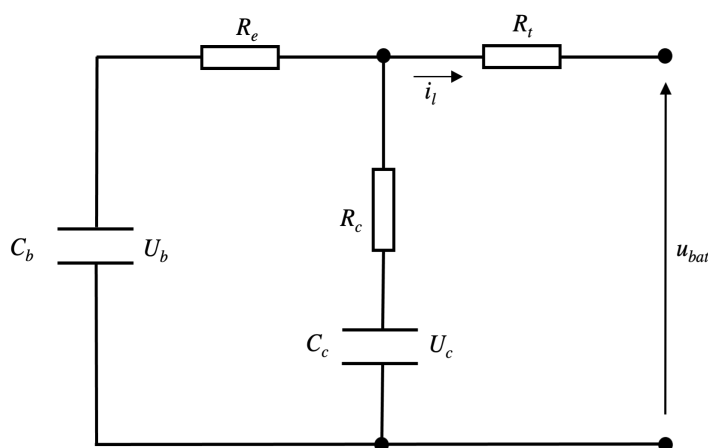
obliku ćelije. Nadalje, dizajn ćelije (debljina aktivnog materijala i slojeva kolektora struje, dodaci elektrolita itd.) može se optimirati s obzirom na gustoću energije ili snage ili za postizanje kompromisa između navedenih karakteristika. Zbog većeg potencijala LTO u odnosu na Li/Li⁺, nominalni napon ćelije je oko 1,2-1,4 V niži u usporedbi s ćelijama s grafitnim anodama. Stoga se više ćelija mora spojiti u seriju kako bi se dobio željeni napon baterije za određenu primjenu. Druga razlika je prošireni temperaturni raspon u kojem ćelije mogu raditi. Moguće je punjenje baterijskih ćelija na temperaturama do -30 °C, što omogućuje punjenje i visokim strujama zimi bez dodatnog grijanja. Brzine struje punjenja koje omogućuju LTO anode jednake su brzini struje pražnjenja, dok je u ćelijama s grafitnim anodama moguća struja punjenja znatno niža od struje pražnjenja. Slika 1 uspoređuje svojstva relevantna za primjenu baterijskih ćelija s grafitnim i LTO anodama te LiFePO₄.

2. Dinamički model baterije

Postoji mnogo matematičkih modela nadomjesnog strujnog kruga za elektrokemijsku bateriju koji su popularan izbor kada nije poželjno uključivati modele elektrokemijskih procesa koji se događaju u bateriji. Cilj takvih modela je prikazati ponašanje baterije pod opterećenjem zamjenjujući matematičke zakone elektrokemije ekvivalentnim električnim strujnim krugovima [14]. Takvi modeli opisuju punjenje i pražnjenje baterije, gubitke, efekte zagrijavanja te starost baterije. Razlog postojanja više modela je jednostavna; ovisi o primjeni. Ako su nam potrebni detalji kako se ponašaju gubici baterije te želimo optimizirati punjenje/pražnjenje, koristit ćemo matematički kompliciranije modele jer oni vjernije opisuju gubitke. Dok u slučaju gdje nam je potreban jednostavan model s kojim možemo napraviti kvalitetnu regulaciju punjenja, možemo upotrijebiti daleko jednostavniji model.

2.1. RC model baterije

RC model baterije razvijen je od tvrtke SAFT. Sastoji se od dva kondenzatora (C_b , C_c) i tri otpornika (R_e , R_c , R_t). Kondenzator C_b ima veoma veliki kapacitet te predstavlja mogućnost baterije za pohranu naboja, nazvan je „Bulk“ kondenzator, dok kondenzator C_c ima malen kapacitet i najčešće opisuje ograničavajuću isporuku struje baterije zbog vremenskih konstanti povezanih s difuzijom materijala i kemijskim reakcijama same baterije. Stanje napunjenosti baterije određuje se preko napona „Bulk“ kondenzatora [15].



Slika 6: RC model baterije

Ponašanje ovog modela opisano je jednadžbama:

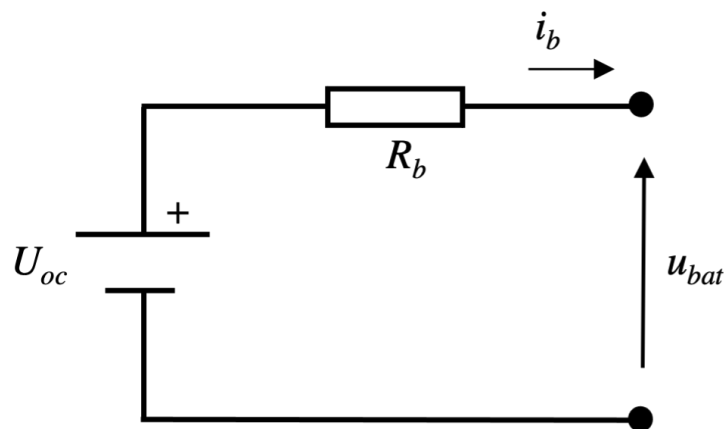
$$\begin{bmatrix} u_{bat} \\ \dot{U}_c \end{bmatrix} = \begin{bmatrix} \frac{-1}{C_b(R_e + R_c)} & \frac{1}{C_b(R_e + R_c)} \\ \frac{1}{C_c(R_e + R_c)} & \frac{-1}{C_c(R_e + R_c)} \end{bmatrix} \begin{bmatrix} u_{bat} \\ U_c \end{bmatrix} + \begin{bmatrix} \frac{-R_c}{C_b(R_e + R_c)} \\ \frac{-R_e}{C_c(R_e + R_c)} \end{bmatrix} [i_l] \quad (2.1)$$

A napon baterije je jednak:

$$[u_{bat}] = \begin{bmatrix} \frac{R_c}{R_e + R_c} & \frac{R_e}{R_e + R_c} \end{bmatrix} \begin{bmatrix} u_{bat} \\ U_c \end{bmatrix} + \begin{bmatrix} -R_t & -\frac{R_e R_c}{R_e + R_c} \end{bmatrix} [i_l] \quad (2.2)$$

2.2. Jednostavni Theveninov model baterije

Jednostavni Theveninov model prikazan je na Slici 7 i opisan jednađbom (2.3). To je jedan od najčešće korištenih, ali i najjednostavnijih modela. Koristi se kod izrade jednostavnih simulacijskih modela baterije. Baterija je prikazana kao serijski spoj idealnog istosmjernog naponskog izvora u_{bat} i unutarnjeg otpora baterije. Pretpostavlja se da su napon otvorenog kruga U_{oc} i unutarnji otpor baterije R_b konstante veličine, što kod realnijih modela nije slučaj, već one ovise o stanju napunjenosti baterije, temperaturi, životnom vijeku baterije.

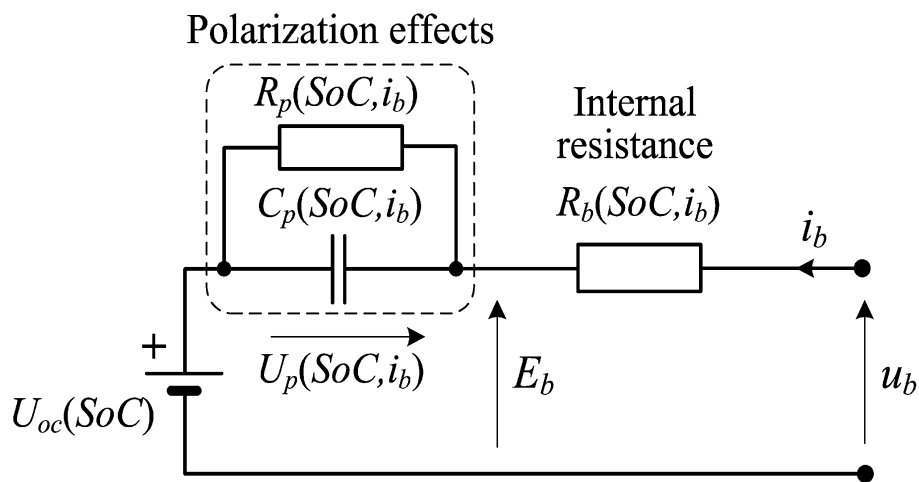


Slika 7: Jednostavni Theveninov baterije

$$u_{bat}(\xi) = U_{oc}(\xi) - R_b \cdot i_b \quad (2.3)$$

2.3. Theveninov model baterije

Često korišteni model u industriji je Theveninov model baterije, sastavljen od napona otvorenog kruga (Napon koji baterija poprima kada je u stanju mirovanja) U_{oc} , unutarnjeg otpora R_b i paralelnog RC spoja polarizacijskog kondenzatora C_p i polarizacijskog otpora R_p . Polarizacijski kondenzator predstavlja kapacitivnost paralelnih ploča baterije kao prijelaznog sloja između ploča i elektrolita te opisuje tranzitne promjene prilikom punjenja odnosno pražnjenja, a ovisi o poroznosti elektroda. Polarizacijski otpor predstavlja nelinearni otpor izazvan kontaktom ploča i elektrolita, a unutarnji otpor predstavlja gubitke baterije [5].

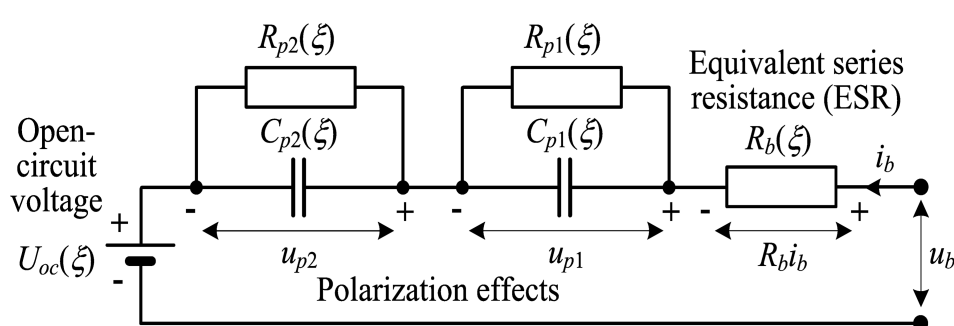


Slika 8: Theveninov model

$$u_b(\xi) = U_{oc}(\xi) + R_b \cdot i_b + u_p \tag{2.4}$$

$$u_p = i_r \cdot R_p$$

Također, Ovakav tip modela možemo proširiti na n broj RC strujnih krugova; koliko točno želimo opisivati polarizacijsku dinamiku i slično te na primjer može sadržavati dvije RC paralelne mreže kao na slici 9.



Slika 9: Theveninov model sa 2 RC strujna kruga

$$u_b(\xi) = U_{oc}(\xi) + R_b \cdot i_b + u_{p1} + u_{p2} \quad (2.6)$$

$$u_{p1} = i_{r1} \cdot R_{p1} \quad (2.7)$$

$$u_{p2} = i_{r2} \cdot R_{p2} \quad (2.8)$$

Ako usporedimo sliku 9 sa slikom 3 vidimo dosta sličnosti u modelima; na isti način opisuju sporu dinamiku i brzu dinamiku, samo što kod modela baterije opisuju polarizacijsku dinamiku. Jedan RC strujni krug opisuje brzu dinamiku (vrijednosti kondenzatora i otpora su manji), dok drugi RC strujni krug opisuje sporu dinamiku. Ovakav model bi bio idealan ako bi nas više zanimalo sami utjecaj polarizacijske dinamike i za izradu kompleksnijih regulacija punjenja. Sami utjecaj polarizacijskih dinamika ne ulazi previše u obzir pri izradi regulatora jer je dosta sporija dinamika (objasnit će se kasnije u radu).

Stoga za daljnji rad ćemo uzeti Theveninov model sa jednim RC strujnim krugom jer dovoljno dobro opisuje potrebnu dinamiku. Također, uzet je model punjenja gdje je pozitivan predznak struje prilikom punjenja, a to će poslužiti kasnije kod dizajna estimatora stanja napunjenosti u sklopu punjača baterije.

2.4. Postavljanje ekvivalentnog strujnog kruga

Pomoću Theveninovog modela opisat ćemo sljedeću dinamiku baterije (slika 8). Za navedeni model će se odraditi set punjenja i pražnjenja (kojeg ćemo kasnije opisati) tako da dobimo dovoljno dobro područje djelovanja baterije. Također, uzet ćemo da se sve odvija pri konstantnoj temperaturi od 25 °C.

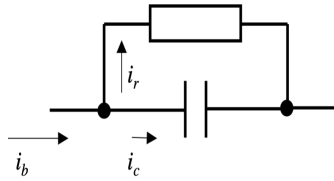
Napon otvorenog kruga (U_{oc}) je napon koji pokazuje vrijednosti napona kada baterija nije spojena na trošilo, te polarizacijski efekti i unutarnji otpor ne igraju veliku ulogu; Taj napon možemo smatrati nominalnim naponom baterije. Napon baterije (u_b) je ukupni napon koji baterija posjeduje prilikom spajanja trošila na nju (Taj napon se promatra pri korištenju same). R_p i C_p utjecaji dolaze do izražaja pri promjeni toka struje ili pri zaustavljanju punjenja/pražnjenja baterije. Također, samo-pražnjenje baterije nismo uzeli u obzir jer ne utječu na samu dinamiku.

Obzirom da je model postavljen kao model punjenja, dobivamo sljedeći izraz:

$$u_b(\xi) = U_{oc}(\xi) + R_b \cdot i_b + u_p \quad (2.9)$$

$$u_p = i_r \cdot R_p \quad (2.10)$$

S obzirom da struja i_r nije ista (ima u sebi kašnjenje zbog postojeće vremenske konstante RC mreže), možemo ga na sljedeći način opisati:



Slika 10: RC strujni krug

$$i_c = C \cdot \frac{dU}{dt} \quad (2.11)$$

Prebacivanjem u s-domenu dobivamo sljedeći izraz:

$$U = \frac{i_c}{C \cdot s} \quad (2.12)$$

Znamo da je napon Otpora jednak kada je kondenzator pun.

Stoga možemo zapisati:

$$\frac{i_c}{C \cdot s} = i_r \cdot R \quad (2.13)$$

Po 1. Kirchhoffovom zakonu znamo da je:

$$i_b = i_c + i_r \quad (2.14)$$

Uvrštavanjem (2.13) i (2.14) formule dobivamo izraz:

$$\begin{aligned} i_b - i_r &= i_r \cdot s \cdot R \cdot C \\ i_b &= i_r \cdot (1 + R \cdot C \cdot s) \end{aligned} \quad (2.15)$$

prebacivanjem članova i uvrštavanjem u (2.10) dobivamo:

$$u_p = \frac{1}{\tau_p s + 1} \cdot i_b \cdot R_p \quad (2.16)$$

Također, stanje napunjenosti baterije ćemo definirati preko električnog naboja te dobivamo daljnji izraz:

$$\xi = \frac{Q_b(s)}{Q_{max}} = \frac{1}{Q_{max}} \frac{i_b(s)}{s} \quad (2.17)$$

Te ako prebacimo u vremensku domenu i uključimo konstantu koja nastaje pritom integriranja, dobivamo sljedeći izraz:

$$\xi = \xi_0 + \frac{1}{Q_{max}(I_b)} \cdot \int_0^t i_b \cdot dt \quad (2.18)$$

Te jednažba napona baterije izgleda:

$$u_b(\xi) = U_{oc}(\xi) + R_b \cdot i_b + \frac{1}{\tau_{ps} + 1} \cdot i_b \cdot R_p \quad (2.19)$$

gdje je $Q_{max}(I_b)$ kapacitet punjenja baterije ovisan o struji, koji se obično modelira kao ovisan o prosječnoj struji baterije i_b [17].

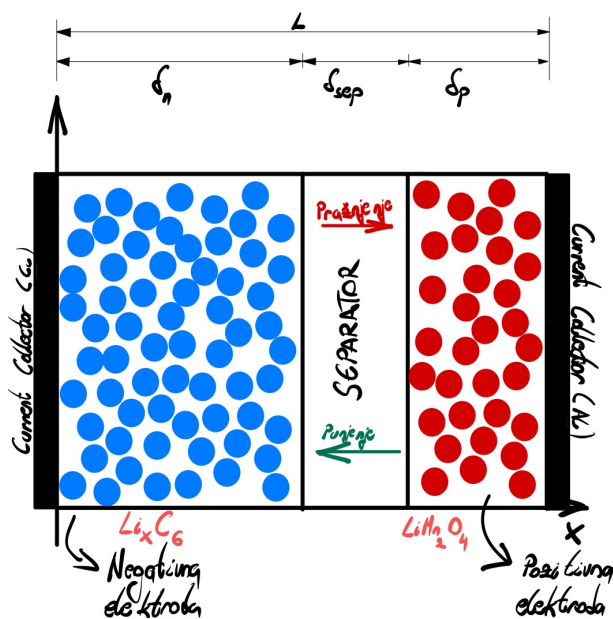
Uz ovako postavljene jednažbe, potrebni su još parametri baterije, točnije krivulja napona otvorenog kruga, unutarnji otpor, polarizacijski otpor te polarizacijski kapacitet. U sljedećoj točki ćemo se posvetiti na određivanje krivulje napona otvorenog kruga, načinom određivanja točaka kod mjerenja (protokol mjerenja) i pristupima nelinearne regresije.

Također, raspisani model ćemo kasnije u poglavlju prebaciti u model prostora stanja koji je potreban za izradu estimatora stanja.

2.5. Termalni model baterije

Kao primjer termalnog modela uzeti su oni koji su razvijeni za baterijske tehnologije Li_xC_6 and LiMn_2O_4 , iako je pristup pri izradi matematičkih modela identičan i za ostale tehnologije zbog sličnih internih struktura uz jedini razliku u materijalima od kojih su interne strukture napravljene. S obzirom da je termalni model veoma bitan u svijetu autoindustrije, počevši od sustava hlađenja pa do sustava sigurnosti pri punjenju baterije. Postoje dva načina kako se mogu modeli odrediti: empirijski i fizikalno. Empirijski modeli su modeli koji se dobivaju primjenom eksperimentalne analize u različitim uvjetima i profilima opterećenja iz razloga da se pokrije što veći spektar stanja. Iz takvih modela, možemo pomoću interpolacije dobivati vrijednosti za koje nismo radili eksperiment, a spadaju u taj spektar. Takav pristup je dosta skup, nepraktičan i dosta je vremenski zahtjevan. Modeli znaju biti dosta kompleksni, a time povlači i pitanje odabira mikrokontrolera (koji može implementirati takav model, na primjer za potrebe balansiranja naboja u ćelijama ili termalnog nadzora na temelju ograničenog skupa mjerenja.). Stoga, postoje i fizikalni modeli koji su daleko jednostavniji i baziraju se na matematičkim jednažbama. Možemo unositi razne pretpostavke (čime štedimo na procesorskoj snazi), nisu nam potrebni stvarni eksperimenti (osim validacije) te je dosta jeftiniji pristup. Jedina mana njih je da im je pristup dosta kompleksniji i postavljanje početnih uvjeta. Stoga, ćemo mi dati

kratak uvod na tu temu, opisati dodatne gubitke te efekte koji se javljaju prilikom punjenja pražnjenja.



Slika 11: Shema litij-ionske ćelije

Na slici 11 su prikazane glavne komponente litij-ionske baterije, uključujući negativni kolektor struje, pozitivni kolektor struje, negativnu elektrodu, pozitivnu elektrodu i separator za blokiranje elektrona. Kolektori negativne i pozitivne struje općenito su izrađeni od bakra i aluminijskog, dok su elektrode i separator napravljeni od poroznog materijala te ispunjeni elektrolitom koji se sastoji od organskog otapala poput etilen karbonata i dimetil karbonata, te litijeve soli $LiPF_6$. Aktivni materijali za negativnu elektrodu i pozitivnu elektrodu su prikazani odvojeno kao Li_xC_6 i $LiMn_2O_4$ na slici 11, dok se debljina negativne elektrode, pozitivne elektrode i separatora označava s δ_n , δ_p i δ_{sep} .

Prema teoriji porozne elektrode, elektrode se mogu podijeliti na čvrstu i elektrolitsku fazu. Tijekom pražnjenja baterije, ioni litija se izlučuju s površine čestica Li_xC_6 , istovremeno se oslobađaju elektroni. Ti ioni litija se potom putem difuzije i migracije kroz separator prenose do pozitivne elektrode, dok elektroni prikupljeni na negativnom kolektoru struje teku kroz vanjski krug kako bi stvorili električnu struju za pražnjenje. Na kraju, litijevi ioni se interkaliraju u čestice $LiMn_2O_4$ zajedno s elektronima koji su dopremljeni iz vanjskog kruga. Tijekom punjenja baterije, transportni procesi litijevih iona i elektrona su obrnuti u odnosu na procese koji se odvijaju tijekom pražnjenja. Ioni litija se izlučuju s površine čestica $LiMn_2O_4$ te se interkaliraju u čestice Li_xC_6 , dok elektroni teku prema negativnoj elektrodi.

Prostorna varijacija parametara kao što su gustoća reakcijske struje i koncentracija aktivnog materijala u elektrodama uglavnom je uzrokovana procesima transporta litijevih iona i elektrona.[21]

Za jednostavniji pristup jednadžbama uvodimo pretpostavke:

- Aktivni materijal je sačinjen od sfernih čestica sa jednakim radijusom te čestice su ravnomjerno raspoređene u elektrodama
- Koeficijent difuzije litij - iona u čvrstom stanju se ne mijenja, stoga se koncentracija litij - iona može promatrati kao konstanta
- Promjene u volumenu i poroznosti su zanemarive
- Ostale reakcije u bateriji su zanemarive

Također, toplina koju razvija ćelija se može klasificirati u 3 dijela:

- Reakcijska toplina
- Toplina unutarnjeg otpora
- Toplina uzrokovana polarizacijskim efektom

Reakcijska toplina spada pod reverzibilnu toplinu dok preostale u ireverzibilnu toplinu. te gubitke možemo zapisati kao:

$$Q_g = Q_{rev} + Q_{irev} \quad (2.20)$$

2.5.1. Reakcijska toplina

Zbog razlike između Gibbsove slobodne energije reaktanta i produkta, dio energije se uvijek želi ili osloboditi ili apsorbirati u obliku topline da sačuva toplinsku ravnotežu cijele reakcije. Reakcijska toplina je ujedno i reverzibilna; može apsorbirati ili oslobađati energiju.

$$Q_R = -S_e \int_0^L \frac{jT_{bat}\Delta S}{n_e F} dx \quad (2.21)$$

Ako preformuliramo da nam samo aktivna površina elektrode ostane unutar integrala, jednadžba izgleda:

$$Q_R = -\frac{S_e j T_{bat} \Delta S}{n_e F} \int_0^L \Delta S dx \quad (2.22)$$

Gdje je ΔS :

$$\Delta S = -\frac{\partial U_{oc}}{\partial T_{bat}} n_e F \quad (2.23)$$

Uvrštavanjem (2.21) u (2.20) i skraćivanjem dobivamo:

$$Q_R = -S_e j T_{bat} \int_0^L \frac{\partial U_{oc}}{\partial T_{bat}} dx \quad (2.24)$$

Integral zapravo predstavlja dužinu od početka do kraja anode i katode, što bi kod ćelije to značilo ukupna dužina od + do -.

Nadalje, možemo malo drugačiji zapis napraviti:

$$Q_R = \frac{\partial U_{oc}}{\partial T_{bat}} T_{bat} \cdot \int_0^L -S_e j dx \quad (2.25)$$

Ovakvim zapisom odvajamo mjereni dio $\frac{\partial U_{oc}}{\partial T_{bat}}$ te dio koji je definiran elektroke-mijom, točnije opisuje struju baterije.

Ako za drugi dio uzmemo pretpostavku da se struja uniformno prenosi onda vrijedi sljedeće:

$$i_{bat} = \int_0^L -S_e j dx \quad (2.26)$$

$$i_{bat} = -S_e j L$$

Stoga zapis reakcijske topline glasi:

$$Q_R = \frac{\partial U_{oc}}{\partial T_{bat}} T_{bat} \cdot i_{bat} \quad (2.27)$$

Ako promatramo reakcijsku energiju zasebno, možemo zaključiti da se za vrijeme punjenja temperatura baterije povećava, dok za vrijeme pražnjenja smanjuje. Taj efekt je dosta osjetan pri malim strujama jer ako pogledamo jednadžbu (2.20), vidimo da utjecaj reverzibilne energije je dosta približan ireverzibilnim.

2.5.2. Toplina unutarnjeg otpora

Elektroni i litijevi ioni bi se sudarali sa transportnim medijem u čvrstoj i elektrolit-skoj fazi tokom transportnog procesa. tako generirana toplina je toplina unutarnjeg otpora. Iako je kompleksno opisati sudaranje elektrona i litijevih iona, možemo zanemariti jedan

dio utjecaja:

$$Q_o = S_e \int_0^L \left[\sigma^{eff} \left(\frac{\partial \phi_s}{\partial x} \right)^2 + \kappa^{eff} \left(\frac{\partial \phi_e}{\partial x} \right)^2 + \kappa_D^{eff} \frac{\partial l_n c_e}{\partial x} \frac{\partial \phi_e}{\partial x} \right] dx + I_b^2 R_b \quad (2.28)$$

Ako bolje uvidimo parametre sa jednadžbe 2.28, možemo vidjeti da član $I^2 R$ je daleko veći po pitanju iznosa te je on nositelj vrijednosti topline. Iz tog razloga, možemo zanemariti prvi dio topline unutarnjeg otpora. Stoga Zapis izgleda:

$$Q_o = i_b^2 R_b \quad (2.29)$$

2.5.3. Toplina uzrokovana polarizacijskim efektom

U stanju kada se baterija ne koristi (Baterija poprima vrijednost napona otvorenog kruga), reakcija deinterkalacije je jednaka interkalaciji, točnije nema prijenosa elektrona, a time ni struje. Kada je ta ravnoteža prekinuta, stvara se struja. Energija koja prekida ravnotežu je toplina uzrokovana polarizacijskim efektom.

$$Q_o = S_e \int_0^L j \cdot (\phi_s - \phi_e - U) dx \quad (2.30)$$

S obzirom da je utjecaj topline kompleksan, model topline ćemo aproksimirati kao 1 RC mrežu kao što je vidljivo na slici 10.

$$Q_p = i_r^2 R_p \quad (2.31)$$

Uzevši u obzir jednadžbe (2.13) i (2.14) dobivamo:

$$Q_p = \frac{i_b^2}{\tau^2 s^2 + 2\tau s + 1} R_p \quad (2.32)$$

Možemo uočiti da model zapravo izgleda kao PT_2 član koji opisuje dinamiku kašnjenja drugog reda.

Ako uvrstimo sve dobivene topline u jednadžbu (2.20) dobivamo gotovi termalni model:

$$Q_G = \frac{\partial U_{oc}}{\partial T_{bat}} T_{bat} \cdot i_{bat} + \frac{i_{bat}^2}{\tau^2 s^2 + 2\tau s + 1} R_p + i_{bat}^2 \cdot R_b \quad (2.33)$$

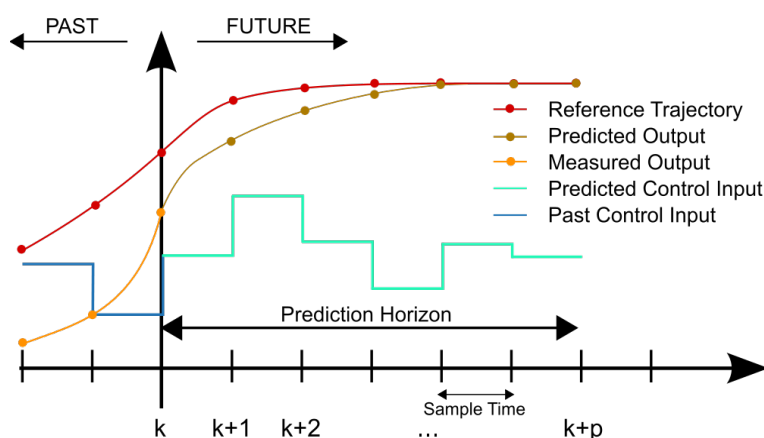
2.6. Uvod u prediktivno upravljanje

Prediktivno upravljanje (engl. Model Predictive Control, MPC) je metoda upravljanja koja se koristi za postizanje optimalnog ponašanja dinamičkih sustava. MPC je matematička upravljačka strategija koja kombinira modeliranje sustava, optimizacijske tehnike i povratnu vezu u stvarnom vremenu kako bi generirala optimalno upravljačko ponašanje.

MPC se koristi u različitim industrijama i primjenama, uključujući procesnu industriju, energetiku, automobilsku industriju, robotsku kontrolu i mnoge druge. Glavna ideja MPC-a je da koristi model dinamičkog sustava kako bi predvidio buduće ponašanje i optimizirao upravljačke akcije u određenom vremenskom horizontu.

Postupak MPC-a uključuje sljedeće korake:

1. **Modeliranje sustava:** Izrađuje se matematički model dinamičkog sustava koji se kontrolira. Model opisuje odnos između ulaznih i izlaznih varijabli sustava.
2. **Prediktivno upravljanje :** Na temelju modela sustava, MPC generira optimalno upravljačko ponašanje za niz budućih vremenskih koraka. To se postiže formuliranjem problema optimizacije koji minimizira zadane kriterije performansi i ograničenja sustava.
3. **Povratna veza:** Trenutna mjerenja sustava koriste se za prilagodbu generiranih upravljačkih akcija kako bi se uzeli u obzir stvarni uvjeti i smanjili učinci pogrešaka modela. Ovaj korak omogućuje prilagodljivo upravljanje u stvarnom vremenu.
4. **Ponavljanje postupka:** Postupak se ponavlja u svakom vremenskom koraku kako bi se generirale nove upravljačke akcije na temelju najnovijih informacija o sustavu.



Slika 12: Princip rada prediktivnog upravljanja

MPC pruža prednosti u odnosu na tradicionalne metode upravljanja, kao što su PID regulatori, jer može rješavati složenije probleme i uzeti u obzir ograničenja sustava i referentnih signala. Također omogućuje postizanje bolje regulacije, brže prilagodbe i optimalne upravljačke akcije.

Važno je napomenuti da je MPC sofisticirana metoda koja zahtijeva dobar matematički model sustava, računalnu snagu i optimizacijske algoritme za rješavanje problema optimizacije u stvarnom vremenu.

Matematički zapis upravljanja možemo rastaviti na dvije funkcije: Ciljna funkcija (Koja je direktno vezana sa sustavom upravljanja i trebala bi težiti u nulu jer penaliziramo grešku sustava) te funkcija troška preko koje optimiziramo sustav na N horizontu.

Ciljna funkcija izgleda:

$$l(x, u) = \|x_u - x^r\|_Q^2 + \|u - u^r\|_R^2 \quad (2.34)$$

Penaliziramo preko druge norme razlika stoga da uvijek imamo pozitivnu grešku.

Funkcija troška izgleda:

$$\min_u J_N(x, u) = \sum_{k=0}^{N-1} l(x_u(k), u(k)) \quad (2.35)$$

Jednadžbe (2.34) te (2.35) opisuju problem optimalnog upravljanja što za naš sustav nije previše poželjno jer zna imati svoje mane za nelinearne sustave. Primjerice, prilikom većeg horizonta N funkcija troška postaje dosta nelinearna što može utjecati na samu optimizaciju. Također, nije prikladna metoda za nelinearne i/ili nestabilne sustave kada se optimizira na veći vremenski horizont.

Stoga problem optimalnog upravljanja bi trebalo prebaciti u problem nelinearnog programiranja koji rješava navedene probleme.

Metoda koja omogućava prebacivanje iz problema optimalnog upravljanja u problem nelinearnog programiranja se naziva *Multiple Shooting*. Metoda je bazirana da se smanji korakintegracije u male vremenske intervale tako da model sistema dobiva ograničenja stanja na svakom optimizacijskom koraku. iz jednadžbe (2.35) varijable odlučivanja za optimizacijski problem postaje i varijabla stanja i upravljačka varijabla te zapis izgleda na sljedeći način:

$$w = \left[u_0, u_1, \dots, u_{N-1}, x_0, x_1, \dots, x_{N-1} \right] \quad (2.36)$$

$$\min_w \phi(w) \quad (2.37)$$

$$\text{subject to: } g_1(w) = \begin{bmatrix} g_1(x_0, u_0) \\ \cdot \\ \cdot \\ g_1(x_{N-1}, u_{N-1}) \\ g_1(x_N) \end{bmatrix} \leq 0, \quad (2.38)$$

$$g_2(w) = \begin{bmatrix} \tilde{x}_0 - x_0 \\ f(x_0, u_0) - x_1 \\ \cdot \\ \cdot \\ f(x_{N-1}, u_{N-1}) - x_N \end{bmatrix} = 0$$

Ovakvim zapisom dobivamo više varijabla, no smanjujemo nelinearnost sustava što upućuje na poboljšanje pri konvergenciji rješenju.

Za postavljanje MPC regulatora, pristup koji bi uzeli bi bio sljedeći; Jednadžbe ograničenja bi bile jednažbe u kojima bi se nalazilo temperaturna ograničenja u kojem bi regulator funkcionirao, ograničenja pretvarača te same baterije. Time bi mogli točno utjecati na ulaznu struju te se osigurati u slučaju nastanka kratkog spoja. kao ulaznu varijablu bi imali struju, a ujedno i upravljačku varijablu jer bi optimirao samu vrijednost. Stanje koje bi nam bilo od velike važnosti za pratiti je stanje napunjenosti baterije te bi točno mogli regulirati na koliko *SoC*-a želimo napuniti, je li nam termalni model zadovoljava visoke struje te je li dosegnoo maksimum struje od strane pretvarača. Također, regulator se može koristiti i za pražnjenje u sličnom okruženju, primjerice u hibridnom vozilu. Ako želimo optimirati potrošnju baterije s time da pazimo da nam ne dosegne na niske granice *SoC*, morali bi imati u set podataka rute preko koje bi mogli računati točno u kojem trenutku vrijedi prazniti bateriju, a u kojem puniti.

Također, treba spomenuti da matematika prediktivnog upravljanja je primjenjiva i u estimaciji parametara, koji se također zasniva na pomičnom horizontu. u poglavlju *Estimator stanja* detaljnije opisujemo primjenu i pristup estimatora s pomičnim horizontom.

3. Identifikacija modela baterije

3.1. ARX metoda

ARX (*AutoRegressive with eXogenous inputs*) metoda je jedna od popularnih metoda za identifikaciju nadomjesnih modela baterije. Ova metoda se temelji na autoregresivnom modeliranju sustava s exogenous inputima, što znači da uzima u obzir prethodne vrijednosti izlaza sustava (u ovom slučaju napona ili struje baterije) i exogenous inpute (kao što su napon punjenja, struja opterećenja itd.) za predviđanje budućih vrijednosti.

ARX metoda koristi sljedeći matematički model:

$$y(t) = -a_1y(t-1) - a_2y(t-2) - \dots - a_p y(t-p) + b_0u(t-d) + b_1u(t-d-1) + \dots + b_q u(t-d-q) \quad (3.1)$$

- $y(t)$ - izlazna vrijednost sustava (npr. napon ili struja baterije) u trenutku t
- a_1, a_2, \dots, a_p - koeficijenti autoregresivnog dijela modela koji predstavljaju utjecaj prethodnih vrijednosti izlaza
- b_0, b_1, \dots, b_q - koeficijenti exogenous dijela modela koji predstavljaju utjecaj exogenous inputa
- $u(t)$ - exogenous input (npr. napon punjenja, struja opterećenja) u trenutku t
- d - vremenski pomak exogenous inputa

Prije početka identifikacije ARX modela, potrebno je prikupiti podatke o bateriji. To uključuje snimanje vremenskih serija podataka o naponu i struji baterije, kao i exogenous inputima kao što su napon punjenja, struja opterećenja i drugi relevantni parametri. Ovi podaci moraju biti uzorkovani s dovoljnom brzinom kako bi se uhvatila dinamika baterije. Sljedeći korak je odabir reda modela, odnosno broj prethodnih vrijednosti izlaza (p) i broj prethodnih exogenous inputa (q) koje će model uzeti u obzir. Odabir reda modela može biti empirijski, temeljen na analizi podataka, ili se može koristiti neka metrika kao što je Akaikeov informacijski kriterij (AIC) ili slično. Nakon odabira reda modela, koristi se metoda najmanjih kvadrata ili metoda najveće vjerojatnosti za procjenu koeficijenata modela koji najbolje odgovaraju podacima. Ova procjena uključuje minimiziranje odstupanja između izmjerenih vrijednosti baterije i vrijednosti koje generira ARX model. Identificirani ARX model se zatim provjerava i testira kako bi se ocijenila njegova prediktivna točnost i sposobnost opisivanja ponašanja baterije. To se obično radi na neovisnim podacima koje model nije koristio za identifikaciju.

3.2. ARMAX metoda

ARMAX (*AutoRegressive Moving Average with eXogenous inputs*) je statistička metoda za identifikaciju i modeliranje dinamičkih sustava koji uzima u obzir autoregresivne, moving average i exogenous komponente. Ova metoda se često koristi za identifikaciju nadomjesnih modela baterija i drugih dinamičkih sistema. Prednost ARMAX metode je njena sposobnost modeliranja i uključivanje exogenous inputa, što omogućuje bolje opisivanje i predviđanje sustava koji su izloženi utjecaju vanjskih faktora. Međutim, ARMAX modeli mogu biti osjetljivi na prisutnost šuma i zahtijevaju dovoljan broj dostupnih podataka za preciznu procjenu koeficijenata. Primjer primjene ARMAX metode u identifikaciji nadomjesnih modela baterija može uključivati prikupljanje podataka o naponskim i strujnim vrijednostima baterije, kao i exogenous inputima kao što su napon punjenja, temperatura, vrijeme punjenja i drugi relevantni parametri. Na temelju tih podataka, ARMAX metoda se koristi za estimaciju koeficijenata i konstrukciju modela koji najbolje opisuje dinamiku baterijskog sustava.

Općenito, ARMAX model je proširenje ARX modela koje omogućuje modeliranje nelinearnosti i explicitno uzima u obzir šum u sustavu. ARMAX model je pogodan za sustave s nelinearnim karakteristikama i kada je važno modelirati utjecaj exogenous inputa na izlaz sustava. ARX model je jednostavniji i prikladan za sustave koji se ponašaju linearno-vremenski. Odabir između ARX i ARMAX metoda ovisi o složenosti sustava, dostupnim podacima i specifičnim zahtjevima identifikacije baterije.

3.3. Hammerstein-Wiener metoda

Hammerstein-Wiener model je matematički model koji kombinira Hammerstein model i Wiener model kako bi se bolje opisale nelinearne i linearno-vremenske dinamičke karakteristike sustava. Ovaj model je često korišten u identifikaciji nadomjesnih modela baterija zbog svoje sposobnosti modeliranja složenih nelinearnih svojstava. Hammerstein model opisuje nelinearnost između exogenous inputa i izlaza sustava. Pretpostavlja se da je izlaz sustava rezultat nelinearne obrade ulaznih signala. Nelinearnost se može modelirati pomoću funkcije nelinearnosti koja uzima exogenous input kao ulaz i generira nelinearni izlaz. Ova funkcija nelinearnosti može biti bilo koja funkcija koja najbolje opisuje specifične nelinearne karakteristike sustava, poput polinomijalnih funkcija, sigmoidnih funkcija, eksponencijalnih funkcija itd. Wiener model, s druge strane, opisuje linearnost između exogenous inputa i izlaza sustava. Pretpostavlja se da izlaz sustava proizlazi iz linearnog procesa koji koristi ulazne signale. Linearnost se može modelirati pomoću impulsnog odziva ili frekvencijske funkcije sustava. Impulsnim odzivom opisuje se kako sustav reagira na impulsni ulaz, dok frekvencijska funkcija daje odnos između

frekvencijskog sadržaja ulaznog i izlaznog signala. Hammerstein-Wiener model kombinira ove dvije komponente kako bi se dobila bolja aproksimacija dinamičkog ponašanja sustava. Nelinearni Hammerstein dio modelira složene nelinearne karakteristike sustava, dok linearni Wiener dio modelira linearno-vremenske komponente. To omogućuje modeliranje i opisivanje kompleksnijih sustava, kao što su baterijski sustavi s nelinearnim karakteristikama.

Opis Hammerstein-Wiener modela preko jednadžbi uključuje nelinearnu Hammerstein komponentu i linearnu Wiener komponentu.

Nelinearnu Hammerstein komponentu možemo opisati sljedećom jednadžbom:

$$y_{hammerstein}(k) = F(x(k)) + e(k) \quad (3.2)$$

- $y_{hammerstein}(k)$ je izlaz Hammerstein komponente u trenutku k ,
- $x(k)$ je exogenous input u trenutku k ,
- $F(\cdot)$ je funkcija nelinearnosti koja modelira nelinearno ponašanje sustava,
- $e(k)$ je nelinearni šum ili pogreška u trenutku k .

Linearnu Wiener komponentu možemo opisati sljedećom jednadžbom:

$$y_{wiener}(k) = G(u(k)) \cdot h(k) + v(k) \quad (3.3)$$

- $y_{wiener}(k)$ je izlaz Wiener komponente u trenutku k ,
- $u(k)$ je exogenous input u trenutku k ,
- $G(\cdot)$ je prenosna funkcija linearnog dijela Wiener komponente,
- $h(k)$ je impulsnog odziv ili frekvencijska funkcija linearnog dijela Wiener komponente,
- $v(k)$ je linearni šum ili pogreška u trenutku k .

Te ukupna jednadžba izgleda:

$$y(k) = y_{hammerstein}(k) + y_{wiener}(k) \quad (3.4)$$

Čime dobivamo kompletni opis linearnih i nelinearnih dijelova modela.

S obzirom da nam karakteristika napona otvorenog kruga baterije (jedna od glavnih nelinearnosti u modelu) dosta slični tangensu, točnije samo je na početku i kraju nelinearan dio, za identifikaciju parametara će se koristiti ARX metoda jer je dovoljno jednostavna, a efikasna jer nema problema sa mogućim šumom.

3.4. ARX model baterije prikladan za identifikaciju

Ako uzmemo jednadžbu (2.9) te uvrstimo i (2.16) dobivamo:

$$u_b(\xi) = U_{oc}(\xi) + R_b \cdot i_b + \frac{1}{\tau_p s + 1} \cdot i_b \cdot R_p \quad (3.5)$$

Pretpostavimo da se napon baterije otvorenog kruga relativno sporo mijenja skupa sa stanjem napunjenosti $\left(\frac{\partial U_{oc}}{\partial \xi} \approx 0, \text{ u području } 20 < \xi < 80\right)$, dinamika struje i baterije se može aproksimirati linearnom prijenosnom funkcijom. Stoga, U_{oc} možemo "maknuti" iz modela tj. oduzeti od u_b jer želimo zadržati perturbaciju linearnog modela uzrokovano PRBS-om (u čemu je sačinjen polarizacijski dio).

$$\Delta u_b(s) = \frac{R_b \tau_p s + (R_p + R_b)}{\tau_p s + 1} \cdot i_b(s) \quad (3.6)$$

Model (3.6) možemo transformirati u diskretno područje pomoću Z-transformacije sa ZOH elementom na ulaz procesa modela (struja baterije i_b) te rezultira sljedeću funkciju u z-domeni:

$$\Delta u_b(z) = \frac{B(z)}{A(z)} i_b(z) = \frac{b_1 z + b_0}{z - a} i_b(z) \quad (3.7)$$

gdje su parametri modela (T je vrijeme uzorkovanja):

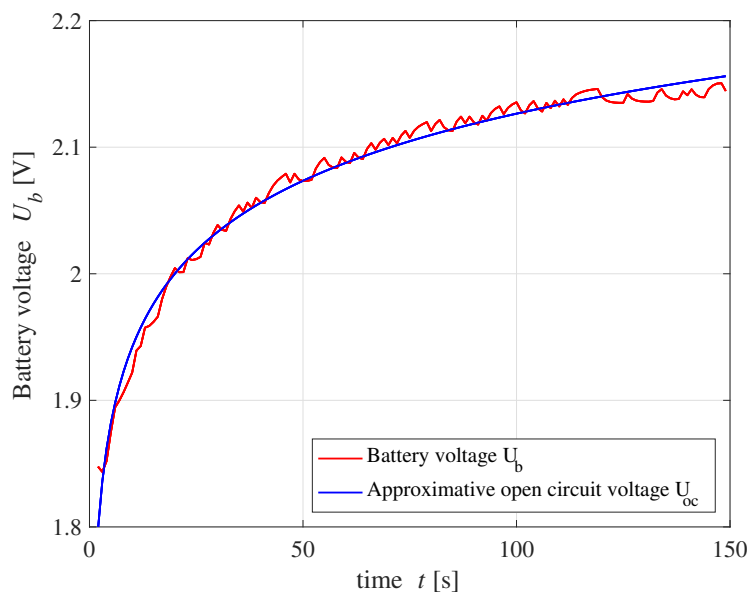
$$a = \exp\left(\frac{-T}{\tau_p}\right), \quad b_1 = R_b, \quad b_0 = R_p - a(R_p + R_b) \quad (3.8)$$

Pod pretpostavkom da se stohastičke perturbacije u mjerenju napona na terminalu baterije mogu modelirati Gausovim šumom u diskretnom vremenu, jednadžba razlike koja opisuje varijaciju napona baterije u odnosu na trenutnu dinamiku može se dati u obliku auto-regresivnog modela s egzogenim ulazima (ARX model) [18]:

$$\Delta u_b(k) = a \Delta u_b(k-1) + b_1 i_b(k) + b_0 i_b(k-1) + \eta(k) \quad (3.9)$$

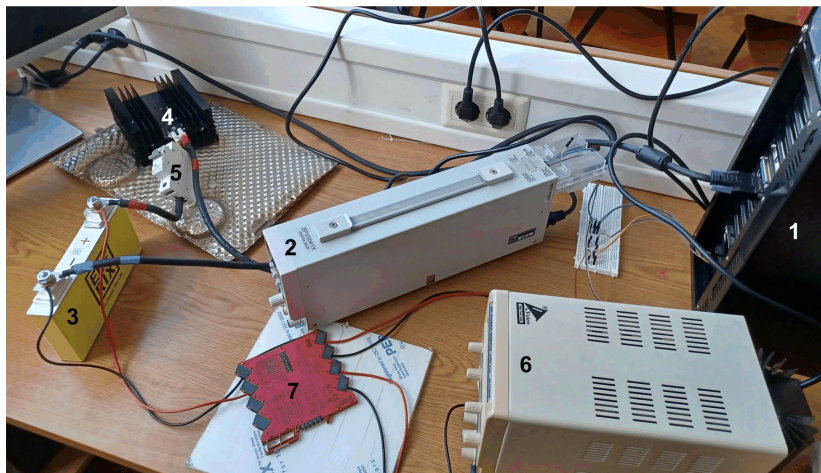
Gdje je k korak uzorkovanja i $\eta(k)$ perturbacija nalik bijelom šumu.

Za nelinearne dijelove (početak i kraj) ćemo umjesto linearizirane prijenosne funkcije, aproksimirati nelinearne dijelove koji nastaju uzorkovanjem PRBS-om. Time dobivamo aproksimativnu funkciju U_{oc} (vidi sliku 13) koju možemo uvrstiti u (3.9).



Slika 13: Napon baterije uzrokovan PRBS-om te aproksimacija napona otvorenog kruga

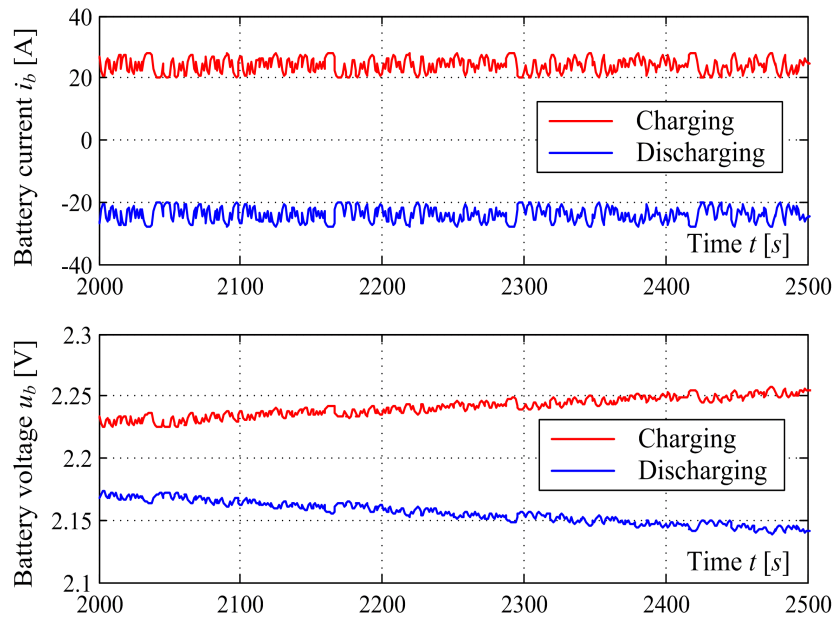
Eksperimentalni postav za testiranje baterije razvijena je u svrhu eksperimentalne karakterizacije, modeliranje i identifikaciju naprednih litijevih baterija, poput onih temeljenih na LiFePO_4 i LTO kemije te i za testiranje regulatora i estimatora stanja/parametara. fotografija razvijene eksperimentalne postavke prikazana je na slici 14. Sastoji se od industrijskog računala (oznaka 1 na slici 14) opremljeno karticama za akviziciju i upravljanje. Industrijsko računalo pokreće namjenski upravljački softver u stvarnom vremenu i šalje odgovarajuće naredbe trenutne reference na laboratorijski DC/DC pretvarač struje (oznaka 2 na slici 14) koji sadrži integriranu funkciju kontrole struje i ograničenja napona, koja se koristi za punjenje i pražnjenje komercijalnu baterijsku ćelije 30Ah/2,4V/6C [16](oznaka 3 na slici 14). Iz operativnih sigurnosnih razloga, strujni krug za punjenje/pražnjenje baterije također ima blokirajuću diodu velike snage (oznaka 4 na slici 14) koja sprječava povratni tok struje uzrokovano neispravnim spajanjem električnih stezaljki i namjenskim prekidačem (oznaka 5 u Slika 14) za zaštitu od prekomjerne struje. Kako bi se izolirao kanal za mjerenje napona baterije na kontrolnoj i akvizicijskoj kartici s terminala napajanja DC/DC pretvarača, postava također ima izolacijsko pojačalo za mjerenje napona (oznaka 7 na slici 14) sa zasebnim napajanjem (oznaka 6 na slici 14).



Legenda: 1 - industrijsko računalo opremljeno karticama za akviziciju i upravljanje; 2 - DC/DC pretvarač; 3 - litij - titanat baterijska ćelija (30 Ah / 2.4 V); 4 - blokirajuća dioda; 5 - Osigurač; 6 - pomoćno 24 V_{DC} napajanje; 7 - izolacijsko pojačalo za mjerenje napona

Slika 14: Postava za testiranje baterije

Snimanje mapa parametara modela ekvivalentnog kruga baterije s obzirom na stanje napunjenosti baterije i struje baterije provedeno je pomoću testova punjenja i pražnjenja baterije, pri čemu je referentna konstantna struja iz DC/DC pretvača uvećana signalom pseudoslučajnog binarnog niza (PRBS) [18] s amplitudom od vrha do vrha od 8 A (tj. ± 4 A). Kako bi se eliminirali učinci *aliasinga* zbog uzorkovanja s relativno velikim vremenom uzorkovanja (ovdje $T = 1$ s), mjerenje napona na terminalu baterije i struja baterije DC/DC pretvarača su preduzorkovani s periodom uzorkovanja od 2 ms i filtrirani sa anti-aliasing niskopropusnim filtrom s propusnošću $\Omega_{bw} = 2 \pi$ rad/s (vidi npr. [20]). Primjer tako dobivenih signala za naknadnu obradu prikazan je na slici 24 za slučaj istosmjerne struje punjenja i pražnjenja od 24 A s prethodno spomenutim PRBS signalom superponiranim na njega (gornja slika na slici 15) kako bi se pobudile male smetnje u naponu baterije (donja slika na slici 15). Razlog primjene PRBS-a je time da tokom cijelog ciklusa punjenja/pražnjenja pobuđujemo polarizacijsku dinamiku.

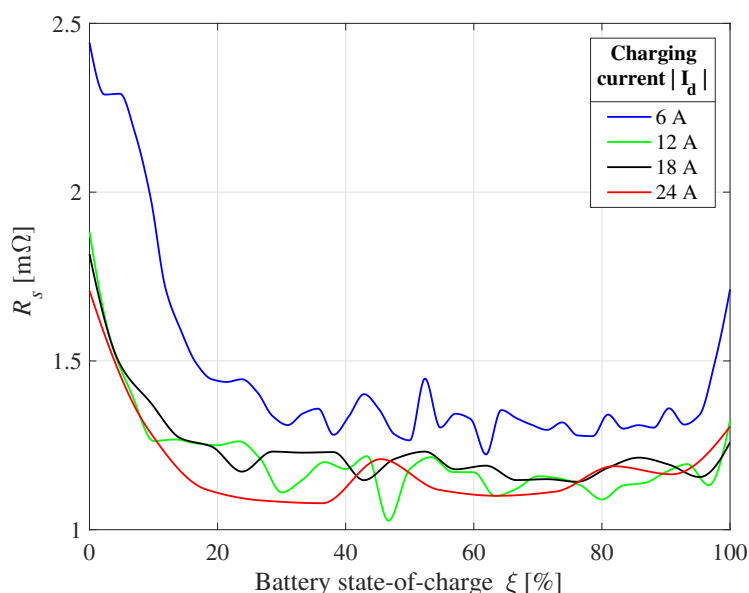


Slika 15: Usporedni eksperimentalni tragovi tijekom punjenja i pražnjenja s prosječnom (DC) strujom od 24 A sa superponiranim perturbacijama PRBS signala od ± 4 A.

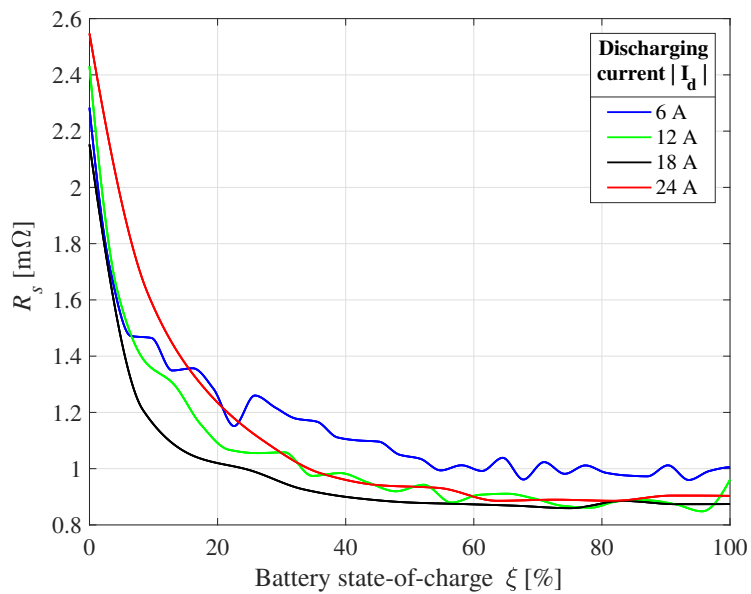
Estimacija parametara prijenosne funkcije baterije izvođena je na sljedeći način:

- Podjelom snimljenih signala struje i napona baterije u skupove jednakih duljina, pri čemu je duljina podataka odabrana tako da bude najmanje dva reda veličine duža od broja parametara koji se procjenjuju (tri parametra u ovom slučaju, R_b , R_p i τ_p).
- Uklanjanje linearnog trenda iz napona terminala baterije koji odgovara postupnom povećanju ili smanjenju napona otvorenog kruga baterije sa stanjem napunjenosti tijekom punjenja ili pražnjenja (vidi sliku 24).
- Procjena linearnog ARX modela u blizini radne točke stanja napunjenosti i prosječne struje baterije korištenjem algoritma procjene najmanjih kvadrata [18].
- Raspored estimiranih parametara baterije u dvodimenzionalne (2D) mape $R_b(\xi, i_b)$, $R_p(\xi, i_b)$ i $\tau_p(\xi, i_b)$ za kasniju upotrebu unutar modela simulacije baterije i estimatora stanja napunjenosti temeljen na metodologiji proširenog Kalmanovog filtra [19].
- Mjerenja odraditi za 6A, 12A, 18A i 24A

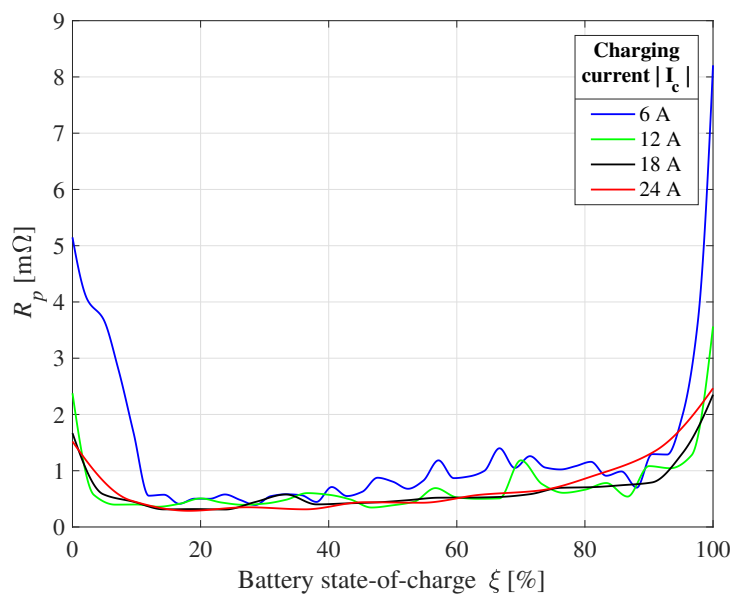
Slike 16 – 21 prikazuju dvodimenzionalne mape parametara ekvivalentnog kruga baterije (Theveninov model) snimljene za razmatranu LTO baterijsku ćeliju od 30 Ah/ 2,4 V [16]. Slika 16 i 17 pokazuje da estimirani unutarnji otpor baterije R_b pokazuje prilično niske vrijednosti u sredini raspona stanja napunjenosti ($R_b < 1,5 \text{ m}\Omega$) za režime punjenja i pražnjenja, ali ima tendenciju porasta do $2,5 \text{ m}\Omega$ kada je baterijska ćelija potpuno ispražnjena ($\xi \rightarrow 0$). Sličan trend je također uočen u slučaju polarizacijskog otpora R_p (slika 18 i 19) čija je vrijednost između $1 \text{ m}\Omega$ i $2 \text{ m}\Omega$ u sredini raspona stanja napunjenosti, dok brzo raste s pražnjenjem baterije. prema nultom stanju naboja (gdje poprima vrijednosti između $4 \text{ m}\Omega$ i $7 \text{ m}\Omega$). Sličan se učinak također opaža u slučaju punjenja prema potpuno napunjenoj bateriji ($\xi \rightarrow 100 \%$). Vremenska konstanta polarizacije τ_p poprima raspon vrijednosti, tj. obično je između 2 s i 20 s, koje uvelike ovise o struji punjenja-pražnjenja i stanju napunjenosti baterije (slika 20 i 21).



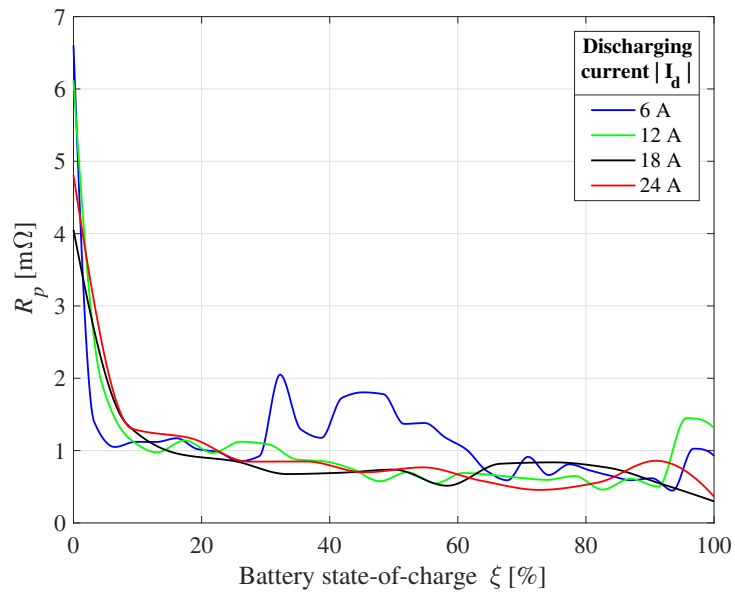
Slika 16: Unutarnji otpor baterije u odnosu sa stanjem napunjenosti - punjenje



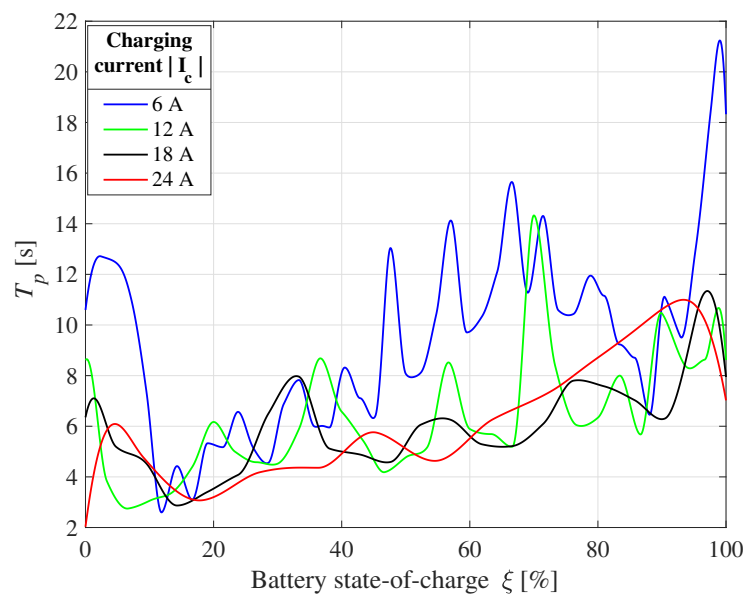
Slika 17: Unutarnji otpor baterije u odnosu sa stanjem napunjenosti - pražnjenje



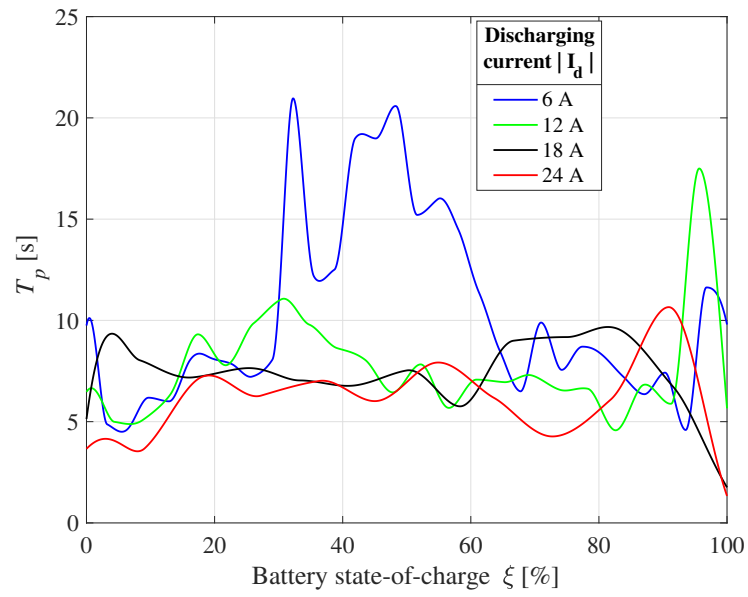
Slika 18: Polarizacijski otpor baterije u odnosu sa stanjem napunjenosti - punjenje



Slika 19: Polarizacijski otpor baterije u odnosu sa stanjem napunjenosti - pražnjenje



Slika 20: Polarizacijska vremenska konstanta baterije u odnosu sa stanjem napunjenosti - punjenje



Slika 21: Polarizacijska vremenska konstanta baterije u odnosu sa stanjem napunjenosti - pražnjenje

3.5. Određivanje krivulje napona otvorenog kruga

Krivulja napona otvorenog kruga baterije se dobiva korištenjem testa punjenja konstantnom strujom (u ovom slučaju sa 6 A). Eksperiment je karakteriziran sa periodičnim kratkotrajnim intervalom punjenja nakon kojog slijedi razdoblje rekuperacije od najmanje dva sata i razdoblja smirivanja napona baterije. Upravo tom rekuperacijom smanjujemo utjecaj polarizacijske dinamike, koja se javlja zbog same kemije baterije, i dobivamo točniju krivulju napona otvorenog kruga.

Protokol mjerenja je sljedeći:

- Od početka punjenja do 20 % SoC-a - punimo po 1 Ah da dobijemo bolju točnost nelinearne karakteristike
- Od 20 % SoC-a do 75 % SoC-a - punimo po 3 Ah jer je ovo područje linearne karakteristike
- Od 75 % do 100 % SoC-a ponovno punimo po 1 Ah jer je ovo isto područje nelinearne karakteristike

Možemo primijetiti jednu zanimljivost kod mjerenja, naime od proizvođača znamo da je baterija deklarirana sa 30 Ah, no mi smo ju nabili do 34 Ah. To je očekivana i normalna stvar jer proizvođač u tolerancijama daje nazivne vrijednosti. U sljedećem dijelu ćemo pokazati 3 tipa nelinearne regresije s kojima ćemo opisati mjerene točke, usporediti sve tri te odabrati pravovaljanu za daljnje modeliranje modela baterije.

Tablica 3: Mjerene vrijednosti napona otvorenog kruga

n_n	U_{oc} [V]	ξ [%]	ΔQ [Ah]
n_0	1,75322	0*	0*
n_1	2,03987	2,941	1
n_2	2,09326	5,882	2
n_3	2,10675	8,824	3
n_4	2,11610	11,765	4
n_5	2,12451	14,706	5
n_6	2,13360	17,647	6
n_7	2,13860	20,588	7
n_8	2,16270	29,412	10
n_9	2,18163	38,235	13
n_{10}	2,19778	47,059	16
n_{11}	2,22238	55,882	19
n_{12}	2,24607	64,706	22
n_{13}	2,29094	73,529	25
n_{14}	2,30815	76,470	26
n_{15}	2,32132	79,412	27
n_{16}	2,34510	82,353	28
n_{17}	2,36715	85,294	29
n_{18}	2,39125	88,235	30
n_{19}	2,41715	91,177	31
n_{20}	2,44550	94,118	32
n_{21}	2,48900	97,059	33
n_{22}	2,68371	100	34

3.5.1. Nelinearna regresija pomoću tangens funkcije

S obzirom da krivulja napona dosta slični tangens funkciji, probat ćemo aproksimirati funkciju sa tangens funkcijom polinomom prvog reda.

Problem:

$$U_{oc}(\xi) = y(x) = a \cdot \tan(bx + c) + y_0 \quad (3.10)$$

Parametre a , b , c i y_0 treba pronaći, a primjenom metode najmanjih kvadrata ćemo ih odrediti u rasponu mjerenih točaka n .

$$S = \sum_{k=1}^n (y_k - a \cdot \tan(bx + c) - y_0)^2 \quad (3.11)$$

Da bismo napravili regresiju i linearizirali, koristit ćemo se integralnim rješenjem

željene funkcije.

$$y(x) = \frac{b}{a} \int (y(x))^2 dx - 2 \frac{bd}{a} \int y(x) dx + \left(\frac{bd^2}{a} + ab \right) \cdot x + C$$

$$y(x) = \frac{b}{a} \sum y^2 \cdot (x_k - x_{k-1}) - 2 \frac{by_0}{a} \sum y \cdot (x_k - x_{k-1}) \quad (3.12)$$

$$S_1 = 0 \rightarrow S_k = S_{k-1} + \frac{1}{2} (y_k + y_{k-1}) (x_k - x_{k-1}), \quad k = 2 \rightarrow n$$

$$T_1 = 0 \rightarrow T_k = T_{k-1} + \frac{1}{2} (y_k^2 + y_{k-1}^2) (x_k - x_{k-1}), \quad k = 2 \rightarrow n \quad (3.13)$$

$$\begin{bmatrix} A \\ B \\ C \\ D \end{bmatrix} = \begin{bmatrix} \sum T_k^2 & \sum T_k S_k & \sum T_k x_k & \sum T_k \\ \sum T_k S_k & \sum S_k^2 & \sum S_k x_k & \sum S_k \\ \sum T_k x_k & \sum S_k x_k & \sum x_k^2 & \sum x_k \\ \sum T_k & \sum S_k & \sum x_k & n \end{bmatrix}^{-1} \begin{bmatrix} \sum T_k y_k \\ \sum S_k y_k \\ \sum x_k y_k \\ \sum y_k \end{bmatrix}$$

↓

$$b = \frac{1}{2} \sqrt{-B^2 + 4AC} = 2.4364 \quad (3.14)$$

$$X_k = \tan(bx_k), \quad k = 1 \rightarrow n$$

Sa sljedećim pojednostavljenjem ćemo odrediti i parametar c :

$$\begin{bmatrix} E \\ F \\ G \end{bmatrix} = \begin{bmatrix} \sum X_k^2 y_k^2 & \sum X_k^2 y_k & \sum X_k y_k \\ \sum X_k^2 y_k & \sum X_k^2 & \sum X_k \\ \sum X_k y_k & \sum X_k & n \end{bmatrix}^{-1} \begin{bmatrix} \sum X_k y_k^2 \\ \sum X_k y_k \\ \sum y_k \end{bmatrix}$$

↓

$$c = \arctan(E) = -1.0315 \quad (3.15)$$

Parametri a i y_0 računamo:

$$\begin{bmatrix} a \\ y_0 \end{bmatrix} = \begin{bmatrix} \sum (\tan(bx_k + c))^2 & \sum \tan(bx_k + c) \\ \sum \tan(bx_k + c) & n \end{bmatrix}^{-1} \begin{bmatrix} \sum \tan(bx_k + c) y_k \\ \sum y_k \end{bmatrix}$$

$$\begin{bmatrix} a \\ y_0 \end{bmatrix} = \begin{bmatrix} 0.1081 \\ 2.1816 \end{bmatrix} \quad (3.16)$$

Te regresija krivulje napona otvorenog glasi:

$$U_{oc}(\xi) = 0,1081 \cdot \tan(2,4364x - 1,0315) + 2,1816 \quad (3.17)$$

3.5.2. Nelinearna regresija pomoću polinomne funkcije

Sljedeći tip nelinearne regresije je dobiven jednostavnije nego prethodna. Pomoću programskog paketa Matlab. Sa dobivenim mjerenim podacima pomoću metode *fit* smo dobili koeficijente polinoma 8. stupnja:

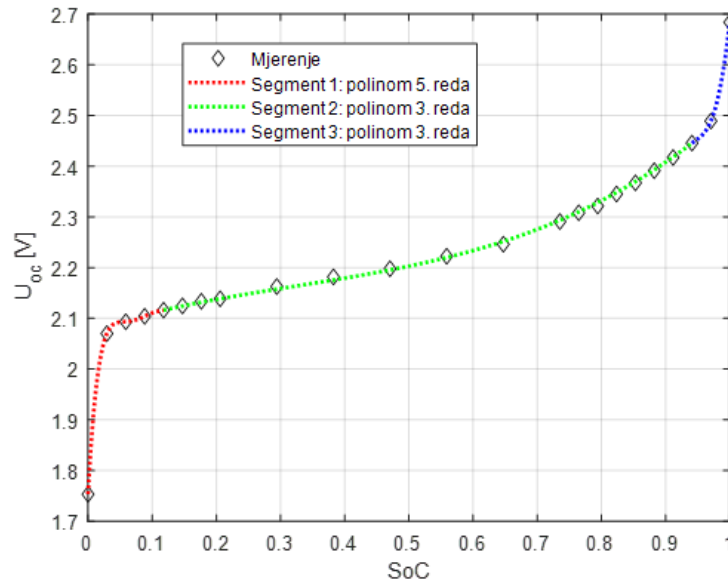
$$U_{oc}(\xi) = \sum_{n=0}^8 p_n \cdot \xi^n \quad (3.18)$$

Tablica 4: Vrijednost polinoma

p_n	Koeficijent polinoma
p_0	$0.0018 \cdot 10^3$
p_1	$0.0090 \cdot 10^3$
p_2	$-0.0861 \cdot 10^3$
p_3	$0.4252 \cdot 10^3$
p_4	$-1.1644 \cdot 10^3$
p_5	$1.8288 \cdot 10^3$
p_6	$-1.6198 \cdot 10^3$
p_7	$0.7409 \cdot 10^3$
p_8	$-0.1325 \cdot 10^3$

3.5.3. Nelinearna regresija pomoću kubičnog spline-a

Dobiveni napon otvorenog kruga u odnosu na točke napunjenosti se aproksimira kroz tri karakteristična radna intervala (nisko stanje napunjenosti, $0 < \xi < 10\%$, srednje stanje napunjenosti $10\% < \xi < 90\%$ i visoko stanje napunjenosti $90 < \xi < 100\%$) kubičnim spline-ovima uz uvjet da gradijent statičke krivulje na granici susjednih intervala mora biti jednak.



Slika 22: Mjerene točke odvojene po segmentima

Ako pogledamo sliku, možemo postaviti potrebne uvjete koji su potrebni za svaki segment.

Segment 1 - prvih pet točaka OCV(ξ) karakteristike:

$$x_1 = [\xi(1) \ \xi(2) \ \xi(3) \ \xi(4) \ \xi(5)]$$

$$y_1 = [U_{oc}(1) \ U_{oc}(2) \ U_{oc}(3) \ U_{oc}(4) \ U_{oc}(5)]$$

Rubni uvjet:

$$\frac{dy_1}{dx_1} = \frac{U_{oc}(6) - U_{oc}(5)}{\xi(6) - \xi(5)}$$

Spline aproksimacija za **segment 1** je uzet polinom petog stupnja:

$$y(x) = p_1(1)x^5 + p_1(2)x^4 + p_1(3)x^3 + p_1(4)x^2 + p_1(5)x + p_1(6) \quad (3.19)$$

S obzirom da znamo vrijednosti x možemo zapisati kao:

$$\begin{bmatrix} x_1(1)^5 & x_1(1)^4 & x_1(1)^3 & x_1(1)^2 & x_1(1) & 1 \\ x_1(2)^5 & x_1(2)^4 & x_1(2)^3 & x_1(2)^2 & x_1(2) & 1 \\ x_1(3)^5 & x_1(3)^4 & x_1(3)^3 & x_1(3)^2 & x_1(3) & 1 \\ x_1(4)^5 & x_1(4)^4 & x_1(4)^3 & x_1(4)^2 & x_1(4) & 1 \\ x_1(5)^5 & x_1(5)^4 & x_1(5)^3 & x_1(5)^2 & x_1(5) & 1 \\ 5x_1(5)^4 & 4x_1(5)^3 & 3x_1(5)^2 & 2x_1(5) & 1 & 0 \end{bmatrix} \begin{bmatrix} p_1(1) \\ p_1(2) \\ p_1(3) \\ p_1(4) \\ p_1(5) \\ p_1(6) \end{bmatrix} = \begin{bmatrix} y_1(1) \\ y_1(2) \\ y_1(3) \\ y_1(4) \\ y_1(5) \\ \frac{dy_1}{dx_1} \end{bmatrix}$$

Segment 2 - od pete do N-2 točke OCV(ξ) karakteristike:

$$x_2 = [\xi(5) \quad \xi(N-2)]$$

$$y_2 = [U_{oc}(5) \quad U_{oc}(N-2)]$$

Spline aproksimacija za **segment 2** je uzet polinom trećeg stupnja:

$$y(x) = p_2(1)x^3 + p_2(2)x^2 + p_2(3)x + p_2(4) \quad (3.20)$$

S obzirom da znamo vrijednosti x možemo zapisati kao:

$$\begin{bmatrix} 3x_2(1)^2 & 2x_2(1) & 1 & 0 \\ x_2(1)^3 & x_2(1)^2 & x_1(1) & 1 \\ x_2(2)^3 & x_2(2)^2 & x_1(2) & 1 \\ 3x_2(2)^2 & 2x_2(2) & 1 & 0 \end{bmatrix} \begin{bmatrix} p_1(1) \\ p_1(2) \\ p_1(3) \\ p_1(4) \end{bmatrix} = \begin{bmatrix} \frac{dy_1}{dx_1} \\ y_1(1) \\ y_1(2) \\ \frac{dy_3}{dx_3} \end{bmatrix}$$

Segment 3 - od N-2 do N-te točke OCV(ξ) karakteristike:

$$x_3 = [\xi(N-2) \quad \xi(N-1) \quad \xi(N)]$$

$$y_3 = [U_{oc}(N-2) \quad U_{oc}(N-1) \quad U_{oc}(N)]$$

Rubni uvjet:

$$\frac{dy_3}{dx_3} = \frac{U_{oc}(N-2) - U_{oc}(N-3)}{\xi(N-2) - \xi(N-3)}$$

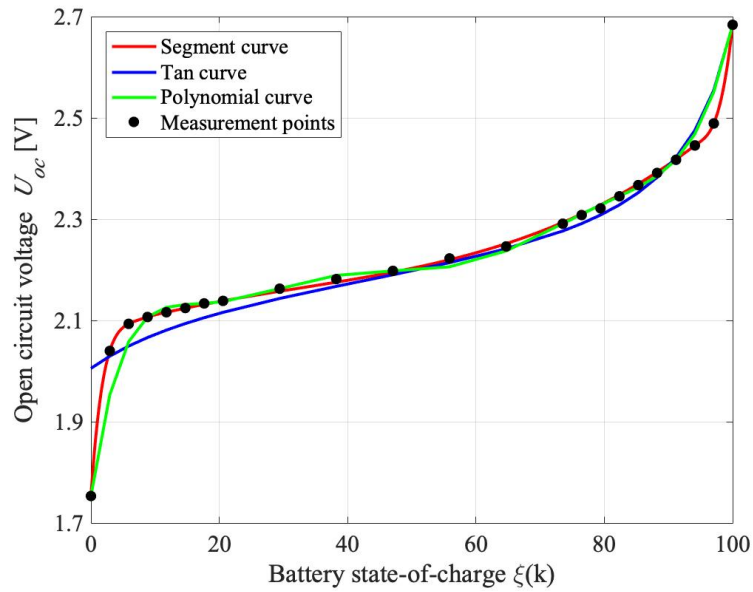
Spline aproksimacija za **segment 2** je uzet polinom trećeg stupnja:

$$y(x) = p_2(1)x^3 + p_2(2)x^2 + p_2(3)x + p_2(4) \quad (3.21)$$

S obzirom da znamo vrijednosti x možemo zapisati kao:

$$\begin{bmatrix} 3x_3(1)^2 & 2x_3(1) & 1 & 0 \\ x_3(1)^3 & x_3(1)^2 & x_3(1) & 1 \\ x_3(2)^3 & x_3(2)^2 & x_3(2) & 1 \\ x_3(3)^3 & x_3(3)^2 & x_3(3) & 1 \end{bmatrix} \begin{bmatrix} p_1(1) \\ p_1(2) \\ p_1(3) \\ p_1(4) \end{bmatrix} = \begin{bmatrix} \frac{dy_3}{dx_3} \\ y_1(1) \\ y_1(2) \\ y_1(3) \end{bmatrix}$$

Pomoću programskog paketa Matlab ćemo izračunati koeficijente te sa dobivenim vrijednostima možemo opisati krivulju.

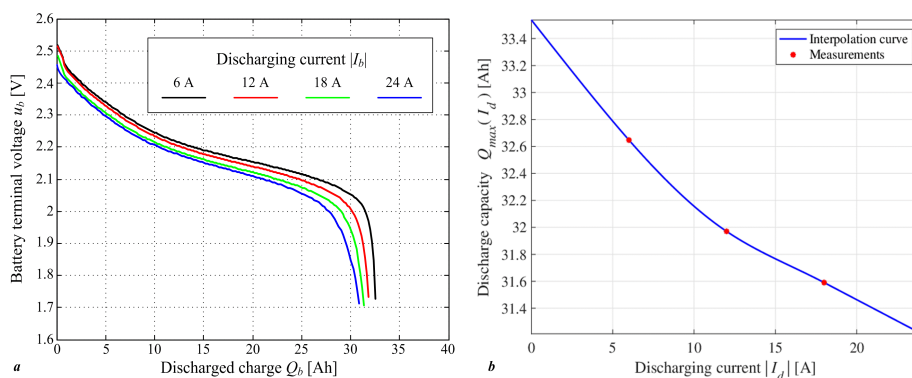


Slika 23: Krivulje napona otvorenog kruga U_{oc}

Ako pogledamo sliku iznad, vidimo regresijske funkcije koje smo računali. Naime, na prvu vidimo na tangens funkcija ne odgovara našoj krivulji U_{oc} -a jer ne prolazi početni nelinearni dio. Polinomna i segmentna funkcija dosta bolje opisuju mjerene točke, no i dalje se vidi da je segmentna konzistentnija od polinomne jer prati vjernije točke. Stoga, za daljnju izradu modela, je uzeta segmentna krivulja.

3.6. Određivanje krivulje nabojskog kapaciteta

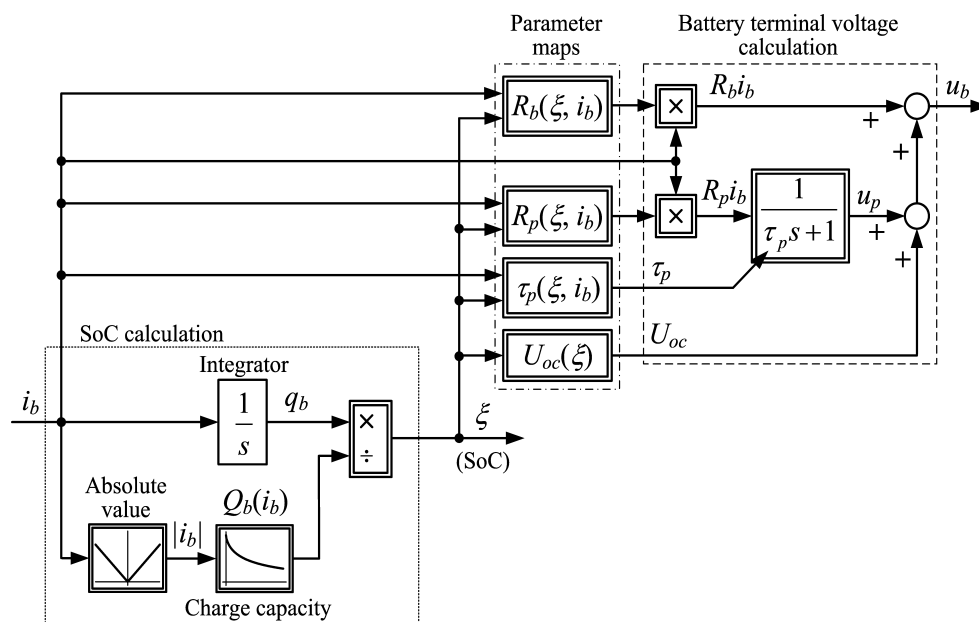
Krivulja kapaciteta baterije u odnosu na struju pražnjenja rekonstruirana je za komercijalnu baterijsku ćeliju od 30 Ah/2,4 V/6C [16]. Rekonstruirana je na temelju testova pražnjenja pri konstantnoj struji, a tako snimljeni naponi terminala baterije u odnosu na krivulje ispražnjenog kapaciteta za konstantne struje pražnjenja prikazane su na slici 24a, pri čemu se eksperiment pražnjenja prekida kada napon ćelije baterije padne ispod 1,7 V. Kapacitet baterije u odnosu na točke struje pražnjenja $Q_b(I_b)$, prikazane su na slici 24b te se interpoliraju preko raspona struje pražnjenja baterije kako bi se naglasio trend kapaciteta punjenja baterije. Rezultati na slici 24b pokazuju da je LTO baterijska ćelija karakterizirana relativno malim smanjenjem kapaciteta punjenja (iznosi manje od 5 %) Q_b u odnosu sa strujom pražnjenja i_b za razmatrani raspon vrijednosti struje pražnjenja (tj. od 6 A do 24 A). Štoviše, procijenjeni kapacitet punjenja neznatno se razlikuje od nominalne vrijednosti od 30 Ah (zapravo je malo veći od nominalne vrijednosti). Gore navedeni rezultati pokazuju da bi se identifikacija kapaciteta punjenja trebala izvršiti za svaku ćeliju prije njezine upotrebe, jer netočna vrijednost kapaciteta punjenja može nepovoljno utjecati na točnost estimacije stanja napunjenosti.



Slika 24: Krivulje pražnjenja konstantne struje baterije (a) i ovisnost kapaciteta baterije u odnosu na struju pražnjenja (b)

3.7. Validacija matematičkog modela baterije

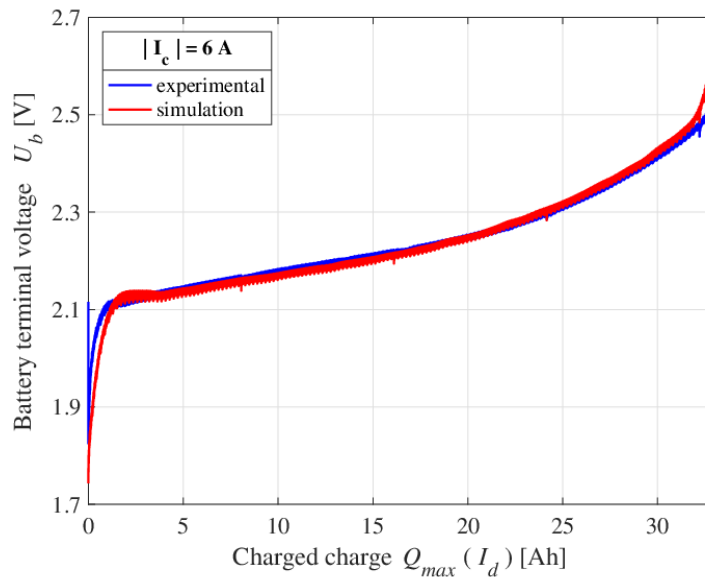
Estimirani parametri ekvivalentnog strujnog kruga baterije koriste se za izradu simulacijskog modela baterije, pri čemu se struja baterije koristi kao ulazna vrijednost modela. Slika 25 prikazuje model kvazistatičke baterije, gdje se akumulirani naboj q_b i polarizacijski napon u_p prema gore tretiraju kao varijable stanja modela. Q_b se koristi za izračunavanje stanja napunjenosti baterije ξ prema jednadžbi (2.18), koristeći formulu kapaciteta napunjenosti baterije ovisno o struji (Sl. 24b). Budući da će reverzibilni elektrokemijski učinci unutar baterije vjerojatno imati sličnu učinkovitost s obzirom na struju baterije i_b za punjenje i pražnjenje, pretpostavlja se da je kapacitet punjenja baterije Q_b isti za pozitivni i negativni predznak struje. Jednadžba napona stezaljki baterije $u_b = R_b i_b + U_{oc} + u_p$ implementirana je pomoću identificiranih statičkih mapa parametara baterije $R_b(\xi, i_b)$, $R_p(\xi, i_b)$, $\tau_p(\xi, i_b)$ i $U_{oc}(\xi)$ kao što je prikazano na slici 25. Cjelokupni simulacijski model i gore spomenute statičke mape parametara implementirane su unutar programskog okruženja Matlab/Simulink.



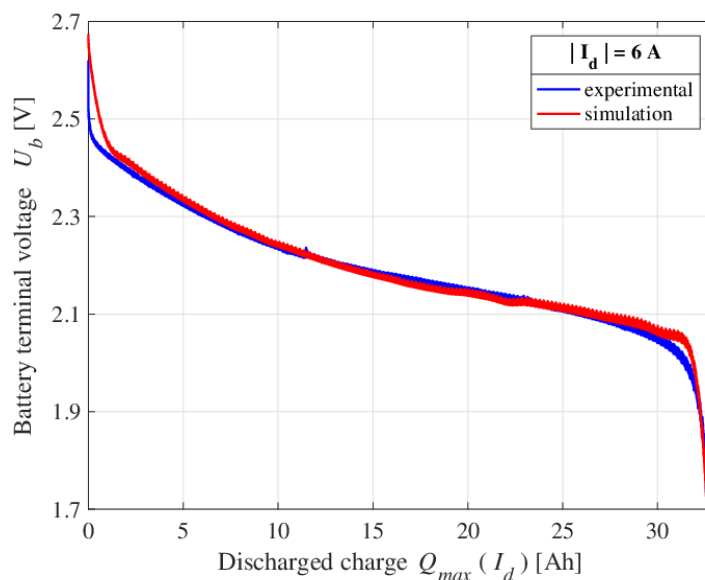
Slika 25: Blok dijagram ekvivalentnog strujnog kruga baterije

Slike 26 – 33 sažimaju usporedne rezultate analize simulacije i odgovarajuće eksperimentalne rezultate dobivene tijekom snimanja podataka korištenih za identifikaciju modela baterije. Rezultati pokazuju da simulacijski model baterije može točno uhvatiti trendove i vrijednosti napona stezaljki baterije u širokom rasponu vrijednosti stanja napunjenosti baterije, s prilično malim odstupanjima uočnim samo za duboko ispražnjenu bateriju ($\xi \rightarrow 0$) i potpuno napunjenu bateriju ($\xi \rightarrow 100\%$). Ova odstupanja mogu se pripisati parametrima baterije koji pokazuju značajne varijacije sa stanjem napunjenosti (osobito unutarnji otpor baterije, polarizacijskog otpora (vidi slike 16-21) i napona

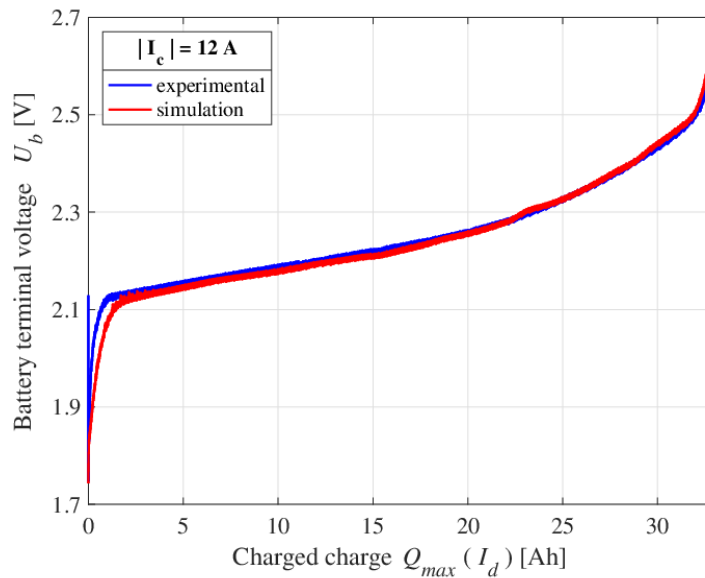
otvorenog kruga). Naime, te bi varijacije bilo teže uhvatiti pomoću eksperimentalne identifikacije linearnog ARX modela, koji pretpostavlja konstantne vrijednosti parametara u blizini radne točke baterije (vidi jednadžbu (3.7)). Usprkos tome, ukupna sposobnost praćenja simulacijskog modela baterije je zadovoljavajuća, stoga se može koristiti za dizajn odgovarajućeg estimatora stanja napunjenosti baterije.



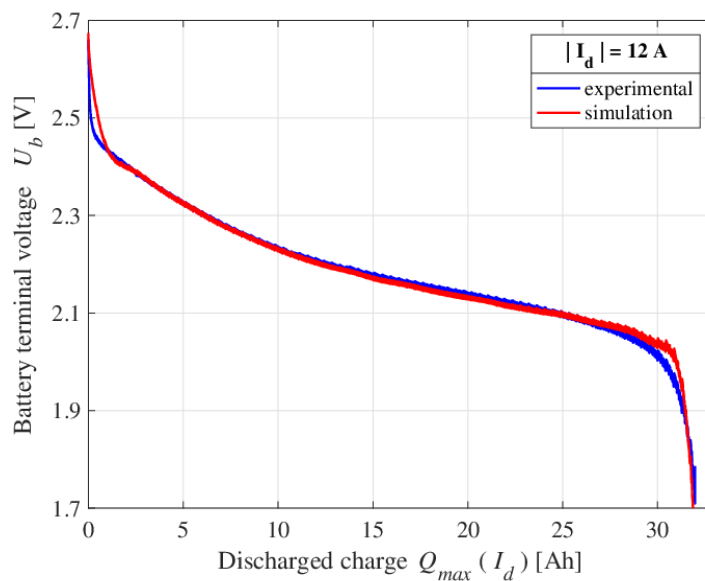
Slika 26: Usporedba simulacije i eksperimentalnog rezultata za napon baterije tijekom punjenja sa strujom baterije od 6 A



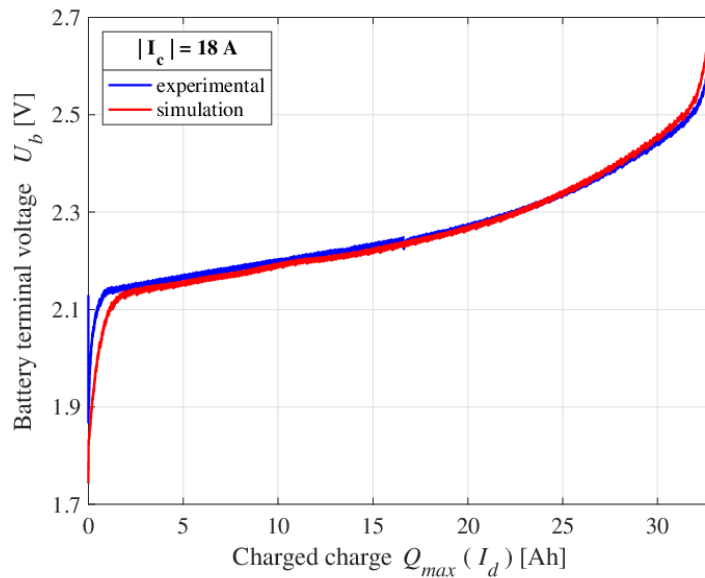
Slika 27: Usporedba simulacije i eksperimentalnog rezultata za napon baterije tijekom pražnjenja sa strujom baterije od 6 A



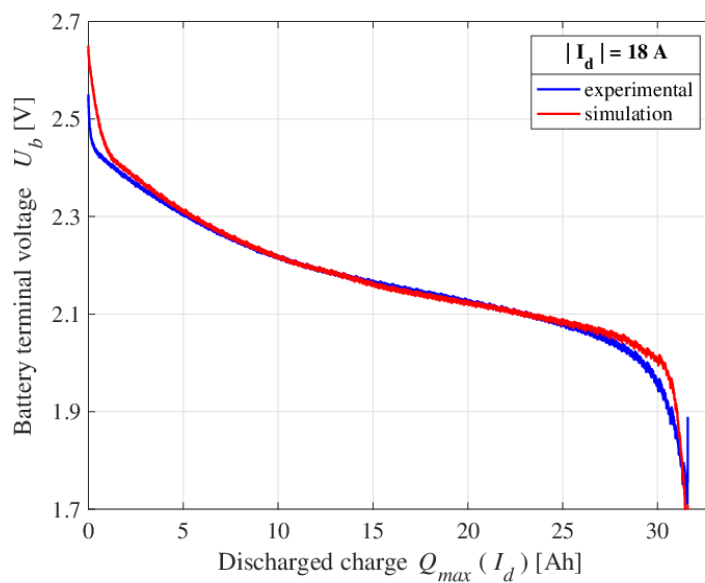
Slika 28: Usporedba simulacije i eksperimentalnog rezultata za napon baterije tijekom punjenja sa strujom baterije od 12 A



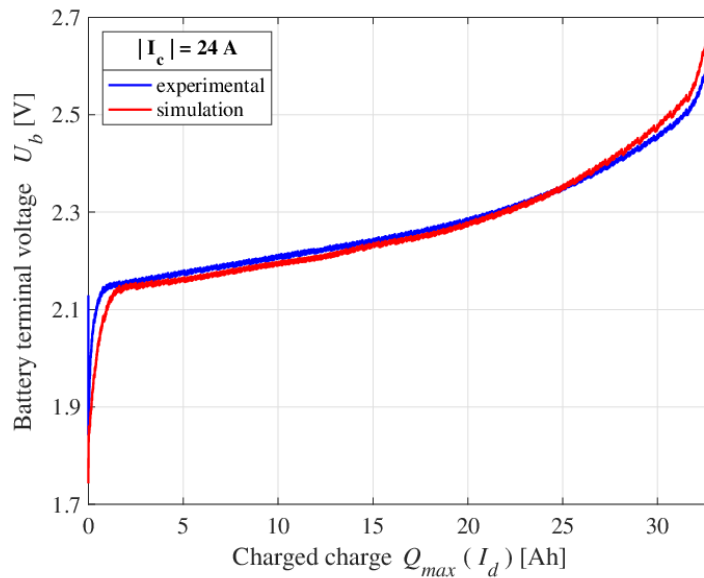
Slika 29: Usporedba simulacije i eksperimentalnog rezultata za napon baterije tijekom pražnjenja sa strujom baterije od 12 A



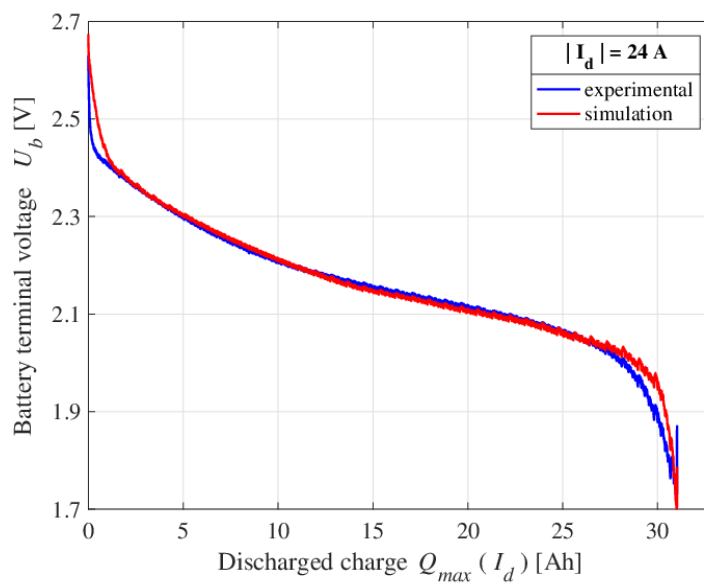
Slika 30: Usporedba simulacije i eksperimentalnog rezultata za napon baterije tijekom punjenja sa strujom baterije od 18 A



Slika 31: Usporedba simulacije i eksperimentalnog rezultata za napon baterije tijekom pražnjenja sa strujom baterije od 18 A



Slika 32: Usporedba simulacije i eksperimentalnog rezultata za napon baterije tijekom punjenja sa strujom baterije od 24 A



Slika 33: Usporedba simulacije i eksperimentalnog rezultata za napon baterije tijekom pražnjenja sa strujom baterije od 24 A

4. Estimator stanja

U ovom poglavlju detaljno su opisani različiti estimatori parametara i varijabli stanja baterije koji su pažljivo odabrani za upotrebu u simulacijama. Njihova uloga je ključna u postizanju pouzdanih rezultata. Sljedeće poglavlje će se posvetiti prezentaciji simulacijskih modela i rezultata dobivenih primjenom ovih estimatorskih tehnika.

Glavni cilj ovog poglavlja je omogućiti dublje razumijevanje rada i funkcioniranja estimatora. Zato su detaljno objašnjeni tri različita tipa estimatora koji su korišteni: Kalmanov filter, prošireni Kalmanov filter (poznat i kao *Extended Kalman Filter*) i estimator s pomičnim horizontom (*Moving Horizon Estimation*)

Kroz njihova opisivanja, upoznat će se s osnovnim principima, matematičkim modelima i praktičnom primjenom svakog od navedenih filtara. Ova cjelovita prezentacija će im pomoći u stjecanju dubljeg uvida u prednosti i ograničenja svakog estimatora te u razumijevanju konteksta u kojem su primijenjeni.

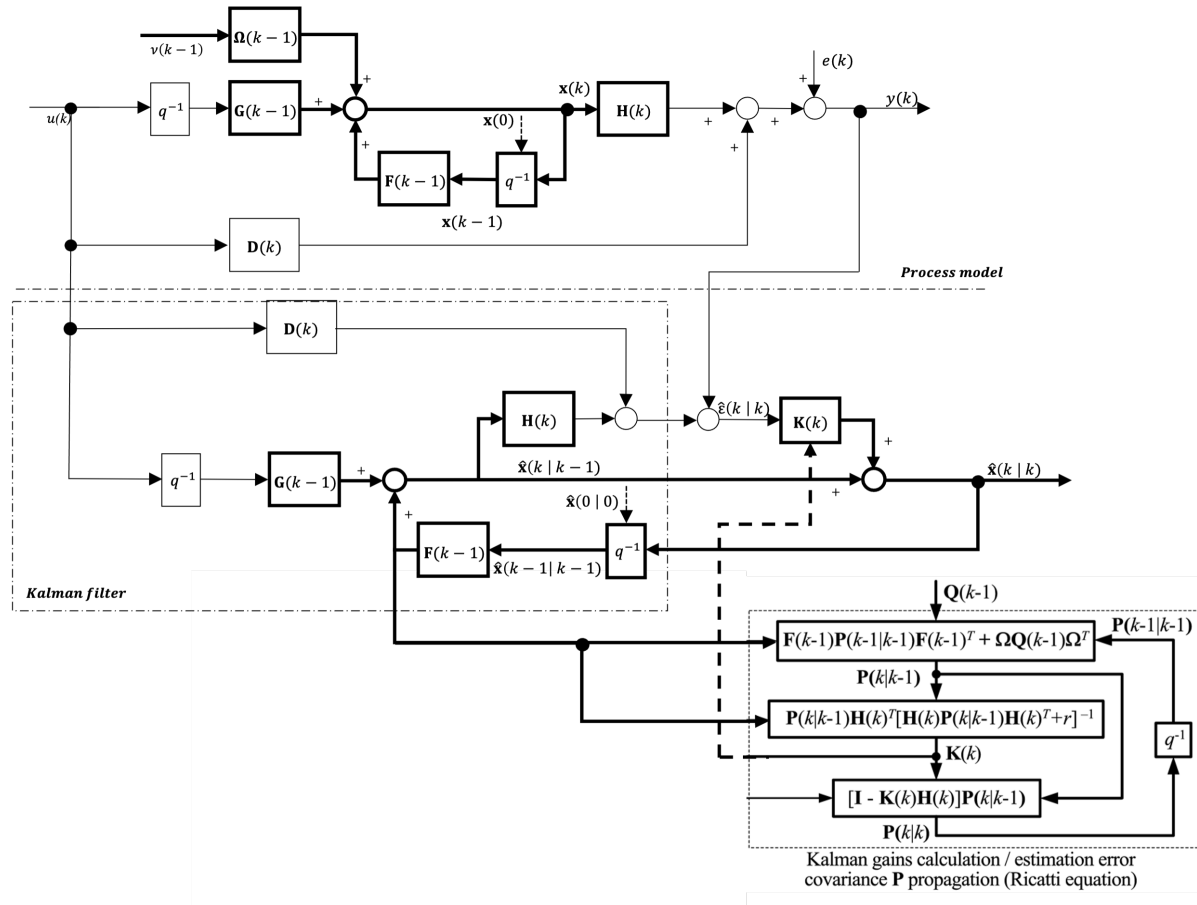
Nadalje, odabrat ćemo jedan od tipova estimatora, dizajnirati ga za naš sustav te prikazati vjernost praćenja samog modela. Također, estimator igra veliku ulogu za izradu neizrazitog regulatora punjača jer je ulazna varijabla stanje napunjenosti baterije, koju ćemo estimirati.

4.1. Kalmanov filter

Kalmanov filter je optimalno rekurzivno rješenje linearnog filtriranja diskretnih podataka koji se najčešće koristi u vojnoj tehnologiji, točnije za autonomno upravljanje i u sustavima za navigaciju. Na samu efikasnost kalmanovog filtara možemo utjecati preko komponenata stohastičkih perturbacija u varijablama stanja $\nu(k)$ i šuma mjerenja $r(k)$ te pretpostavljamo da su oni međusobno neovisni. Također, navedeni vektori predstavljaju matrice kovarijanci (\mathbf{Q} i \mathbf{R}). Njihove vrijednosti ne možemo dobiti mjerenjem te ih sami postavljamo kao konstantnu vrijednost ili promjenjivu (Adaptivni Kalmanov filter).

Primjerice, Ako nismo dovoljno sigurni u naš model, povećat ćemo vrijednosti kovarijance matrice \mathbf{R} jer one djeluju kao mjerene vrijednosti dinamičkog sustava. U slučaju da nismo sigurni da nam mjerene vrijednosti nisu sasvim točne, a sigurni smo da nam je model točan, povećat ćemo vrijednosti matrice kovarijance \mathbf{Q} . Vidjet ćemo u našem primjeru za razne vrijednosti kako nam se model ponaša te donesti zaključak na temelju toga. Upravo takvim aktivnim mijenjanjem vrijednosti \mathbf{Q} i \mathbf{R} postizemo adaptivni Kalmanov filter koji pokušava optimizirati proces uz smanjenje kvadratne greške. Važno je napomenuti da je estimatoru potrebno postaviti točne početne uvjete da može brzo i efikasno funkcionirati.

Također, vrijeme uzorkovanja (T) bi trebalo bit daleko manje (brži estimator) u odnosu na vrijeme uzorkovanja stvarnog modela, tako da estimirane vrijednosti budu precizne stvarnim. S obzirom da je naš model nelinearan, ne možemo primijeniti standardan Kalmanov filtar, stoga koristimo prošireni Kalmanov filtar. Ukratko, to je nelinearni model standardnog Kalmanovog filtra koji koristi linearizaciju oko estimacije stanja i kovarijance.



Slika 34: Kalman filter kao estimator stanja

Zapis jednadžbi izgleda:

$$\mathbf{x}(k) = \mathbf{F}(k-1)\mathbf{x}(k-1) + \mathbf{G}(k-1)u(k-1) + \mathbf{\Omega}(k-1)\boldsymbol{\nu}(k-1) \quad (4.1)$$

$$y(k) = \mathbf{H}(k)\mathbf{x}(k) + \mathbf{D}(k)u(k) + e(k)$$

Matrice $\mathbf{F}(k-1)$, $\mathbf{G}(k-1)$, $\mathbf{H}(k)$ i ulazno-izlazno pojačanje $\mathbf{D}(k)$ modela u prostoru stanja imaju identičnu formu kao i kod determinističkog modela procesa korištenog u sintezi Luenbergerovog estimatora. Sustav je i dalje linearan, samo ima vremenski promjenjive parametre (Kalmanov filtar je izvorno izveden za taj slučaj). Dodatne razlike su

u linearnom aditivnom članu $\mathbf{\Omega}(k-1)\boldsymbol{\nu}(k-1)$ koji opisuje stohastičke perturbacijske komponente u jednadžbe stanja te šumu mjerenja $e(k)$ na izlazu modela procesa.

Pomoću Ricattiijeve jednadžbe dobivamo željeno pojačanje koje nam zapravo pokriva grešku između estimirane vrijednosti i stvarne. Jednadžbe koje se koriste za izračun stanja su sljedeće:

$$\begin{aligned}\hat{x}(k|k-1) &= \mathbf{F}(k-1)\hat{x}(k-1|k-1) + \mathbf{G}(k-1)u(k-1) \\ \hat{e}(k|k-1) &= y(k) - \hat{y}(k|k-1) = y(k) - [\mathbf{H}(k)\hat{x}(k|k-1) + \mathbf{D}(k)u(k)]\end{aligned}\quad (4.2)$$

$$\hat{x}(k|k) = \hat{x}(k|k-1) + \mathbf{K}(k)\hat{e}(k|k-1)$$

Napisane jednadžbe su temeljene na determinističkom dijelu modela procesa.

Preostale jednadžbe su proračun korekcijskih pojačanja optimalnih s obzirom na značajke stohastičkih perturbacija u stanjima i šuma mjerenja:

$$\mathbf{P}(k|k-1) = \mathbf{F}(k-1)\mathbf{P}(k-1|k-1)\mathbf{F}(k-1)^T + \mathbf{\Omega}(k-1)\mathbf{Q}(k-1)\mathbf{\Omega}(k-1)^T$$

$$\mathbf{K}(k) = \frac{\mathbf{P}(k|k-1)\mathbf{H}(k)^T}{\mathbf{H}(k)\mathbf{P}(k|k-1)\mathbf{H}(k)^T + r(k)}$$

$$\mathbf{P}(k|k) = [\mathbf{I} - \mathbf{K}(k)\mathbf{H}(k)] \mathbf{P}(k|k-1)\quad (4.3)$$

4.2. Prošireni Kalmanov filter

Prošireni Kalmanov filter (*Extended Kalman Filter*, EKF) je varijacija Kalmanovog filtra koja se koristi za estimaciju stanja sustava kada je model sustava nelinearan. Dok se klasični Kalmanov filter primjenjuje samo na linearnim sistemima, EKF omogućuje estimaciju stanja i za nelinearne sisteme. EKF koristi linearnu aproksimaciju nelinearnog modela sustava i mjerenja koristeći Taylorov red. Time se omogućuje primjena standardnih Kalmanovih jednadžbi za filtriranje stanja i kovarijanca, ali uz dodatak koraka linearizacije. Kroz iterativni proces, EKF linearno procjenjuje stanje sustava na temelju prethodne procjene i mjerenja, a zatim koristi nelinearnu transformaciju za ažuriranje procjene na sljedećem koraku. Ovaj proces se ponavlja kako bi se postigla što bolja estimacija stanja i kovarijanca, uzimajući u obzir mjerenja i model nelinearnosti sustava.

Jednadžbe koje opisuju EKF su sljedeće:

$$\begin{aligned} \mathbf{x}(k) = & \mathbf{x}_0(k) + \mathbf{F}(k-1) (\mathbf{x}(k-1) - \hat{\mathbf{x}}(k-1|k-1)) + \\ & \mathbf{G}(k-1)u(k-1) + \mathbf{\Omega}(k-1)\boldsymbol{\nu}(k-1) \end{aligned} \quad (4.4)$$

$$y(k) = \mathbf{g}(\mathbf{x}_0(k), 0) + \mathbf{H}(k)(\mathbf{x}(k) - \mathbf{x}_0(k))\boldsymbol{\Psi}(k)\mathbf{e}(k)$$

Jakobijani (\mathbf{F} , \mathbf{H} , $\mathbf{\Omega}$, $\boldsymbol{\Psi}$) za linearizirani model se definiraju kao:

$$\begin{aligned} \mathbf{F}(k-1) &= \left. \frac{\partial \mathbf{f}(\hat{\mathbf{x}})}{\partial \hat{\mathbf{x}}} \right|_{\hat{\mathbf{x}}(k-1|k-1), u(k-1)} \\ \mathbf{H}(k-1) &= \left. \frac{\partial \mathbf{h}(\hat{\mathbf{x}})}{\partial \hat{\mathbf{x}}} \right|_{\hat{\mathbf{x}}(k|k-1), u(k)} \\ \mathbf{\Omega}(k-1) &= \left. \frac{\partial \mathbf{f}(\mathbf{v})}{\partial \mathbf{v}} \right|_{\hat{\mathbf{x}}(k-1|k-1), u(k-1)} \\ \boldsymbol{\Psi}(k-1) &= \left. \frac{\partial \mathbf{h}(\mathbf{e})}{\partial \mathbf{e}} \right|_{\hat{\mathbf{x}}(k|k-1) i_b(k)} \end{aligned} \quad (4.5)$$

Ove matrice lineariziranog sustava potrebne su za proračun optimalnih korekcijskih pojačanja u okolini radne točke estimatora. Korekcijska pojačanja i deterministički dio procesa se računa isto kao i za Kalmanov filter, preko (4.2) i (4.3).

4.3. Estimator s pomičnim horizontom

Moving Horizon Estimation (MHE), također poznat kao receding horizon estimation, je napredna tehnika za procjenu stanja dinamičkih sustava koja kombinira prednosti optimizacijskih metoda i kontrole u stvarnom vremenu. Ova metoda je posebno korisna u situacijama gdje je potrebno procijeniti stanje sustava na temelju nepotpunih, zašumljenih ili ograničenih mjerenja.

MHE se temelji na principu kretajućeg horizonta, što znači da se samo ograničen broj najnovijih podataka koristi za procjenu stanja sustava. Ovaj pristup omogućuje da se procjena prilagodi promjenama u sustavu i osigurava brzu reakciju na promjene u uvjetima okoline.

Prije primjene MHE-a, potrebno je razviti matematički model koji opisuje ponašanje sustava. Ovaj model obično uključuje diferencijalne jednadžbe koje opisuju dinamiku sustava. Odabire se duljina horizonta, tj. broj prethodnih ulaznih i izlaznih podataka koji će se uzeti u obzir prilikom procjene stanja (vidi sliku 35). Ovisno o dostupnim podacima i brzini promjena u sustavu, duljina horizonta može varirati. Procjena stanja se formulira kao optimizacijski problem koji ima za cilj minimizirati razliku između stvarnih izlaza sustava i izlaza generiranih na temelju procjene stanja. Ograničenja se također mogu dodati kako bi se uzela u obzir fizička ograničenja sustava. Rješavanje optimizacijskog problema uključuje iterativni proces u kojem se prilagođava procjena stanja i traži optimalno rješenje. Ovisno o složenosti sustava i zahtjevima vremena, mogu se koristiti različite metode optimizacije. Nakon svake iteracije, novi podaci se koriste za ažuriranje procjene stanja, a proces se ponavlja iznova. Time se osigurava kontinuirano praćenje i procjena stanja sustava u stvarnom vremenu.

Prednosti *Moving Horizon Estimation* (MHE) su višestruke. Ova metoda omogućuje preciznu procjenu stanja dinamičkih sustava čak i kada su dostupni podaci nepotpuni ili šumoviti. Također, MHE ima sposobnost uključivanja ograničenja sustava i prilagođavanja promjenjivim uvjetima. Osim toga, MHE je pogodan za primjene u kojima je potrebno brzo i pouzdano procijeniti stanje sustava, kao što su autonomna vozila, robotska manipulacija, upravljanje energetske sustavima i mnoge druge industrijske primjene.

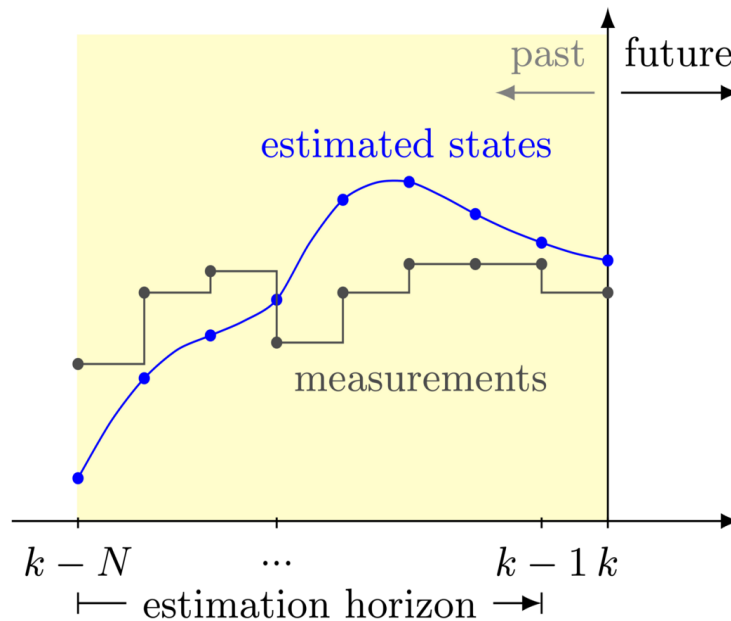
Postavljanje jednadžbe bi glasilo:

$$\begin{aligned} \mathbf{x}(k+1) &= \mathbf{f}(\mathbf{x}(k), u(k)) + \boldsymbol{\nu}_x \\ y(k) &= \mathbf{h}(\mathbf{x}(k)) + \boldsymbol{\nu}_y = \mathbf{x}(k) + \boldsymbol{\nu}_y \end{aligned} \quad (4.6)$$

Pogrešku estimacije u prostoru stanja minimiziramo pomoći sljedeće funkcije kvalitete (engl. cost function):

$$\min_{x,u} J_N(x, u) = \sum_{i=k-N}^k \|\tilde{y}(i) - h(x(i))\|_v^2 + \sum_{i=k-N}^{k-1} \|\tilde{u}(i) - u(i)\|_w^2 \quad (4.7)$$

$$\begin{aligned} \text{s.t. } x_u(i+1) &= \mathbf{f}(\mathbf{x}_u(i), u(i)) \\ u(i) &\in \mathbf{U}, \forall i \in [k - N_{MHE}, k - 1] \\ x_u(i) &\in \mathbf{X}, \forall i \in [k - N_{MHE}, k] \end{aligned}$$



Slika 35: Estimator pomičnog horizonta

Uzevši u obzir sve tipove estimatora, najbolji za našu primjenu bi bio prošireni Kalmanov filter; sustav je moguće linearizirati te je svakako jednostavniji način nego estimator pomičnog horizonta. Obična forma Kalmanovog filtera ne dolazi u obzir jer ne pokriva nelinearne sustave. Stoga, u sljedećem poglavlju ćemo dizajnirati estimator na temelju EKF-a.

4.4. Estimator stanja napunjenosti baterije na temelju proširenog Kalmanovog filtra

Rezultati identifikacije modela ekvivalentnog kruga baterije koriste se kao osnova za estimaciju varijable stanja baterije, što se, prema jednadžbama (2.10) i (2.18), prevodi u on-line procjenu napona polarizacije baterije i stanja napunjenosti. Zbog nelinearne prirode modela baterije (tj. karakteriziran je nelinearnim mapama parametara), prikladan je nelinearni estimator stanja, kao što je prošireni Kalmanov filter (EKF) [19].

U tu svrhu, nelinearni dinamički model baterije prepisuje se u sljedećem obliku prostora stanja:

$$\begin{aligned} \dot{u}_p &= \frac{1}{\tau_p(\xi, i_b)} (R_p(\xi, i_b)i_b - u_p) \\ \dot{\xi} &= \frac{i_b}{Q_b(i_b)} \\ u_b &= u_p + i_b R_b(\xi, i_b) + U_{oc}(\xi) \end{aligned} \quad (4.8)$$

Gornji model se zatim diskretizira u vremenu (uzorkuje) korištenjem Eulerove aproksimacijske vremenske razlike vremenskih derivacija i proširuje se sa Gausovim izvorima šuma (stohastičke perturbacije), što rezultira sljedećim diferencijskim jednadžbama:

$$\begin{aligned} u_p(k) &= \left(1 - \frac{T}{\tau_p(\xi(k-1), i_b(k-1))}\right) u_p(k-1) + \frac{T R_p(\xi(k-1), i_b(k-1))}{\tau_p(\xi(k-1), i_b(k-1))} i_b(k-1) \\ &\quad + T \nu_p(k-1) \\ \xi(k) &= \xi(k-1) + \frac{T}{Q_b(i_b(k-1))} + T \nu_\xi(k-1) \\ u_b(k) &= u_p(k) + i_b(k) R_b(\xi(k), i_b(k)) + U_{oc}(\xi(k)) + e(k) \end{aligned} \quad (4.9)$$

gdje je k korak uzorkovanja, T je vrijeme uzorkovanja, ν_p i ν_ξ su stohastičke perturbacije u varijablama stanja sustava, a e je stohastički mjerni šum u mjerenju napona baterije u_b .

Gornje jednadžbe prikladne su za implementaciju procesa u diskretnom prostoru stanja u okviru EKF estimatora stanja [19]:

$$\begin{aligned} \mathbf{x}(k) &= \mathbf{f}(\mathbf{x}(k-1), i_b(k-1)) + \mathbf{\Omega} \nu(k) \\ u_b(k) &= \mathbf{h}(x(k), i_b(k)) + e(k) \end{aligned} \quad (4.10)$$

s vektorom stanja $\mathbf{x}(k)$ i matricom skaliranja stohastičke perturbacije Ω definirano na sljedeći način:

$$\mathbf{x}(k) = \begin{bmatrix} u_p(k) \\ \xi(k) \end{bmatrix}, \quad \Omega = \begin{bmatrix} T & 0 \\ 0 & T \end{bmatrix}$$

Model baterije je zapisan kao parametar stanja, a s obzirom da smo radili u prostoru stanja potrebno ga je prebaciti. Kako ćemo raditi zapis sa korakom i sa prethodnim korakom, možemo zapisati model kao:

$$\begin{aligned} \mathbf{x}_k &= \mathbf{F} \cdot x_{k-1} + \mathbf{G} \cdot u_{k-1} \\ \mathbf{y}_k &= \mathbf{H} \cdot x_{k-1} + \mathbf{D} \cdot u_{k-1} \end{aligned} \quad (4.11)$$

Matrice \mathbf{F} , \mathbf{G} i \mathbf{H} su zapravo matrice \mathbf{A} , \mathbf{B} i \mathbf{C} prostora stanja, ali su diskretizirane sa vremenom T . Matrice \mathbf{F} i \mathbf{G} mijenjaju formu dok $\mathbf{H} \equiv \mathbf{C}$.

Matrice \mathbf{F} i \mathbf{G} dobivamo na sljedeći način:

$$\mathbf{F} = e^{\mathbf{A} \cdot T} \quad (4.12)$$

$$\mathbf{G} = \mathbf{A}^{-1} \cdot (e^{\mathbf{A} \cdot T} - 1) \cdot \mathbf{B} \quad (4.13)$$

Nadalje, jednačbe (2.16), (2.18) i (2.19) će nam poslužiti za prebacivanje u prostor stanja tako da su nam varijable stanja sljedeće:

$$\mathbf{x}_k = \begin{bmatrix} u_p \\ \xi \end{bmatrix} \quad \mathbf{y} = \begin{bmatrix} u_b \end{bmatrix} \quad \mathbf{u} = \begin{bmatrix} i_b \end{bmatrix} \quad (4.14)$$

Ako deriviramo (2.18) dobivamo sljedeći izraz:

$$\dot{\xi} = \frac{1}{Q_{max}(I_b)} i_b \quad (4.15)$$

Kombinacijom jednačbama (2.12), (2.13) i (2.14) možemo izvesti derivirajući

član polarizacijskog napona.

$$u_p = \frac{i_c}{C \cdot s}, \quad i_r = \frac{u_p}{R}, \quad i_c = i_b - i_r \quad (4.16)$$

$$\begin{aligned} u_p &= \frac{i_b - i_r}{C \cdot s} \\ u_p &= \frac{i_b - \frac{U_p}{R}}{C \cdot s} \\ u_p \cdot C \cdot s &= -\frac{u_p}{R} + i_b \end{aligned} \quad (4.17)$$

Te dobivamo izraz koji se veže za matricu **A** i **B**:

$$u_p = -\frac{1}{R \cdot C} \cdot U_p + \frac{1}{C} \cdot i_b \quad (4.18)$$

Matrice **C** i **D** dobivamo iz jednadžbe:

$$u_b = \xi \cdot U_{oc} + R_b \cdot i_b + u_p \quad (4.19)$$

Te matrice **A**, **B**, **C** i **D** glase:

$$\mathbf{A} = \begin{bmatrix} -\frac{1}{R_p \cdot C_p} & 0 \\ 0 & 0 \end{bmatrix} \quad \mathbf{B} = \begin{bmatrix} \frac{1}{C} \\ 1 \\ \frac{1}{Q_{max}} \end{bmatrix} \quad \mathbf{C} = \begin{bmatrix} 1 & U_{oc} \end{bmatrix}$$

$$\mathbf{D} = \begin{bmatrix} R_b \end{bmatrix}$$

Sa jednadžbama (4.12) i (4.13) i matricama **F** i **G** Dobivamo:

$$\mathbf{F} = \begin{bmatrix} -\frac{1}{R_p \cdot C_p} + 1 & 0 \\ 0 & 1 \end{bmatrix} \quad \mathbf{G} = \begin{bmatrix} \frac{T}{C} \\ \frac{T}{Q_{max}} \end{bmatrix}$$

Matrice **F** i **H** su Jacobijeve matrice, koje se odnose na parcijalne derivacije svih parametara modela baterije s obzirom na stanje napunjenosti baterije i napon polarizacije. Matrica **F** ne zahtijeva linearizaciju (ne sadrži u sebi nelinearnost), no matricu **H** moramo

linearizirati zbog utjecaja U_{oc} :

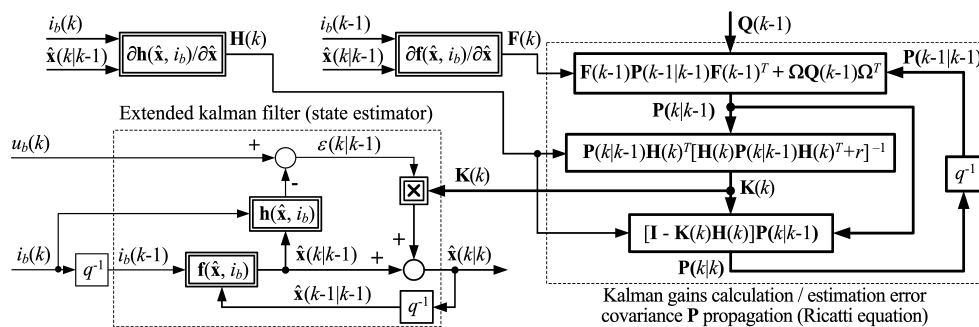
$$\mathbf{H} = \left. \frac{\partial \mathbf{h}(\hat{\mathbf{x}}, i_b)}{\partial \hat{\mathbf{x}}} \right|_{\hat{\mathbf{x}}(k|k-1) i_b(k)} \quad (4.20)$$

Te poprima oblik:

$$\mathbf{H} = \begin{bmatrix} 1 & \frac{\partial U_{oc}(\xi)}{\partial \xi} \end{bmatrix} \quad (4.21)$$

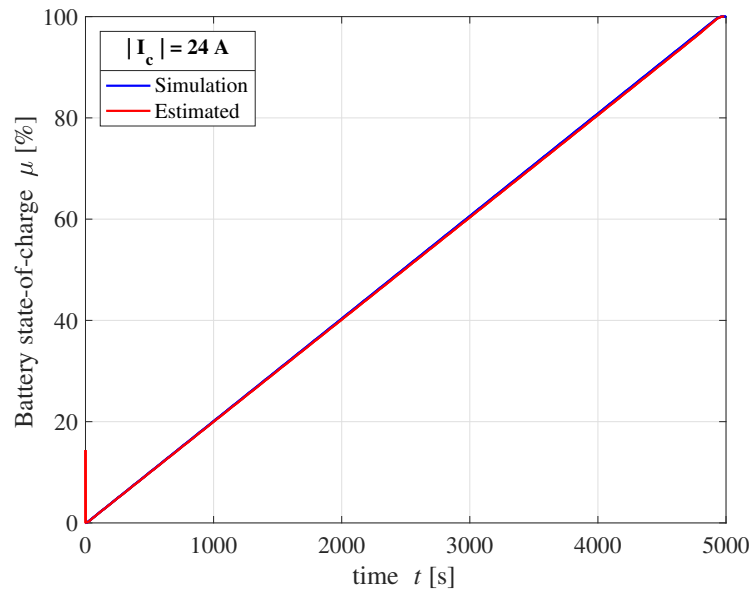
Matrice \mathbf{F} i \mathbf{H} su potrebne za procjenu matrice kovarijance pogreške \mathbf{P} unutar estimatora korištenjem pretpostavljenih varijanci poremećaja stanja ν (karakteriziran matricom kovarijance poremećaja čiji su elementi odabrani kao kompromis između osjetljivosti estimatora na šum i sposobnosti praćenja) i šuma e varijanca r u naponu terminala baterije (mjerenje) u_b .

Prikaz blok dijagrama estimatora temeljenog na EKF-u prikazan je na slici 36, s nelinearnim modelom prostora stanja koji se koristi unutar njega za izračunavanje takozvane apriorne procjene stanja $\hat{\mathbf{x}}(k|k-1)$, koja se naknadno ispravlja putem mjerenja napona baterije što rezultira konačna (a-posteriori) procjena stanja $\hat{\mathbf{x}}(k|k)$. U tu svrhu algoritam estimacije treba izračunati korekcije estimatora na temelju takozvane Ricattijeve jednadžbe [19].

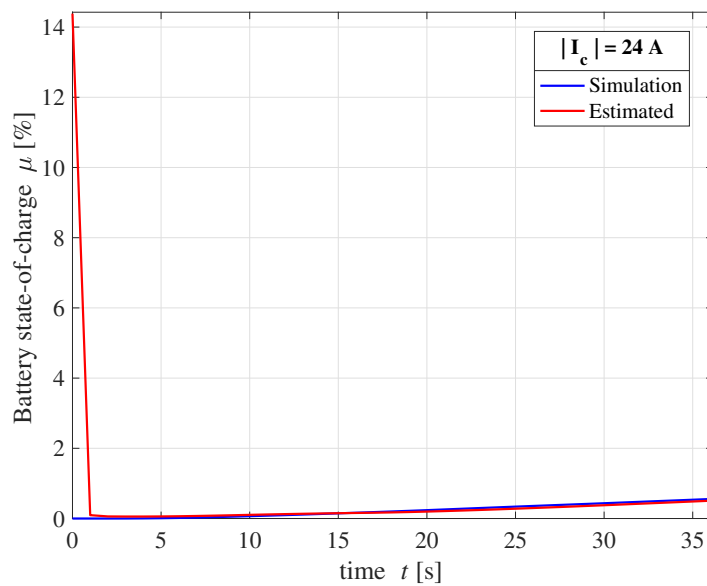


Slika 36: Blok dijagram estimatora stanja

Slika 37 i slika 38 prikazuje rezultate SoC estimatora temeljenog na EKF-u (vidi npr. reference [8] i [9]) podvrgnutog scenariju simulacije punjenja baterije konstantnom strujom sa strujom baterije jednakom 24 A. Rezultati pokazuju prilično dobro poklapanje između stanje napunjenosti dobiveno iz simulacijskog modela i estimatora u stabilnom stanju estimatora (Slika 37), što ukazuje na povoljnu sposobnost praćenja estimatora. Slika 38 pokazuje da je estimator karakteriziran brзом konvergencijom prema stvarnom stanju napunjenosti baterije iz neusklađenog stanja, tj. može se podesiti za prilično brz i dobro prigušen odziv.



Slika 37: Usporedba estimirane i simulacijske vrijednosti stanja napunjenosti tokom punjenja konstantnom strujom



Slika 38: Uvećani dio početnog dijela

5. Dizajn sustava upravljanja

Ovaj poglavlje predstavlja konvencionalnu strukturu sustava upravljanja punjenja (kao referentni slučaj) i predloženi dizajn upravljačkog sustava temeljen na neizrastitoj logici, zajedno s estimatorom stanja napunjenosti baterije temeljenim na estimatoru stanja tipa proširenog Kalmanovog filtra, koji se koristi za pružanje SoC povratne informacije za kontroler punjenja temeljen na neizrastitoj logici.

5.1. Općenito o sustavima PI upravljanja

PI regulator, ili proporcionalno-integrirajući regulator, je vrsta povratne veze (feedback) u sustavu upravljanja koja se koristi za održavanje stabilnosti i preciznosti. Glavna svrha PI regulatora je minimiziranje pogreške regulacije između željene vrijednosti i stvarne izmjerene vrijednosti sustava.

PI regulator kombinira dva glavna elementa: proporcionalni i integracijski (integralni) kontroler. Proporcionalni dio regulira izlaz sustava proporcionalno pogrešci regulacije, dok integracijski dio akumulira pogrešku regulacije tijekom vremena i pruža dugoročno regulaciju sustava. Ova kombinacija omogućuje brzu reakciju na promjene i precizno praćenje željene vrijednosti, uz minimiziranje dugotrajne pogreške.

Metode određivanja pojačanja regulatora u PI regulatoru mogu se provesti na nekoliko načina:

1. Metoda ručnog podešavanja: Ova metoda uključuje eksperimentalno podešavanje regulatora. Prvo se postavlja proporcionalno pojačanje (K_p) na neku početnu vrijednost, a zatim se postavlja integracijsko pojačanje (K_i) na 0. Postupno se povećava K_p dok se sustav ne postigne stabilnost, a zatim se polako povećava K_i kako bi se smanjila dugotrajna pogreška. Ova metoda zahtijeva iskustvo i eksperimentiranje kako bi se postigla željena regulacija.
2. Ziegler-Nichols metoda: Ova metoda je heuristička metoda za određivanje pojačanja regulatora. Metoda se sastoji od nekoliko koraka. Prvo se postavlja K_i na 0, a zatim se postavlja K_p na vrijednost koja je dovoljno velika da uzrokuje oscilacije sustava. Zatim se postavlja kritična pojačanja (K_c) na vrijednost pri kojoj sustav počinje oscilirati, te se izmjeri period oscilacija (T_c). Na temelju ovih vrijednosti, može se izračunati optimalno pojačanje regulatora (K_p) i vremensku konstantu integracijskog dijela (T_i) prema određenim formulama koje su dio Ziegler-Nichols metode.
3. Optimum dvostrukog odnosa: Optimum dvostrukog odnosa, također poznat kao

Optimum Lambda, je koncept u regulacijskoj teoriji koji se odnosi na odabir optimalnog odnosa između proporcionalnog i integracijskog djela u PI regulatoru. Osnovna ideja Optimuma dvostrukog odnosa je da se integracijsko pojačanje povećava smanjenjem pogreške regulacije, ali se također smanjuje kako bi se izbjeglo prekomjerno reguliranje ili oscilacije sustava kada se pogreška približi nuli. To znači da se veće vrijednosti K_i koriste za velike pogreške, dok se manje vrijednosti K_i koriste kada je pogreška blizu nule. Prednosti Optimuma dvostrukog odnosa uključuju brži odziv sustava na promjene, smanjenje oscilacija i poboljšanu stabilnost regulacije. Ovaj pristup omogućuje bolje prilagođavanje regulatora različitim uvjetima rada i varijabilnosti u sustavu.

Optimum dvostrukog odnosa je najbolji izbor za ovakav sustav jer točno možemo utjecati na njegovu dinamiku punjenja; točnije, uz samo promjenu D_n članova, dobivamo sustav brži/sporiji, agresivniji/robusniji.

5.2. Konvencionalni sustav upravljanja punjenja – referentni slučaj

Referentni regulator, u ovom slučaju PI, će biti zasnovan na Optimumu Dvostrukog Odnosa. To je metoda postavljanja polova za linearno invarijantne sustave koristeći se karakterističnom polinomnom jednadžbom čijeg stupnja odgovora stupnju upravljanog sustava.

$$A_c(s) = D_2^{n-1} D_3^{n-2} \cdots D_n T_e^n s^n + \cdots + D_2 T_e^2 s^2 + T_e s + 1 \quad (5.1)$$

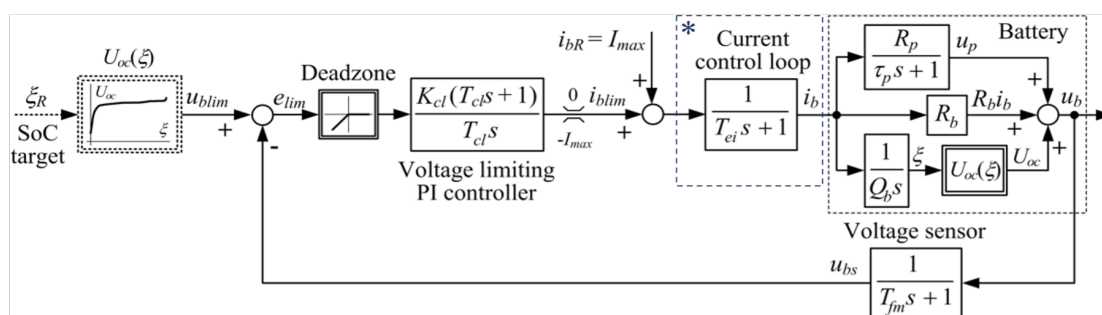
Uz također da su parametri $D_i = 0.5$ jer s tim vrijednostima dobivamo 6% prebačaja i vrijeme dostizanja željene vrijednosti je $(1.8 - 2.1) \cdot T_e$. Za veći T_e sustav je sporiji, no otporniji na smetnje i robusniji odziv posjeduje. Naime, uz pretpostavku da je promjena napona otvorenog kruga vrlo mala te i polarizacijska dinamika, ostaje nam samo unutarnji otpor baterije kao brza promjenjiva dinamika koja znatno može utjecati na regulator.

Slika 39 prikazuje elegantnu strukturu upravljačkog sustava za punjenje baterije konstantnom strujom/konstantnim naponom (CCCV) temeljenu na unutarnjoj strujnoj kontrolnoj petlji. Pretpostavlja se da je ova petlja ugrađena unutar DC/DC pretvarača snage. Uz to, vanjski povratni krug ograničava napon terminala baterije i koristi proporcionalno-integralni (PI) regulator s povratnom spregom. Ovaj napredni sustav kontrole punjenja može se smatrati vrhunskim primjerom prema referencama [8] i [10].

U ovom iznimnom rasporedu kaskadnog upravljanja, unutarnja petlja upravljanja

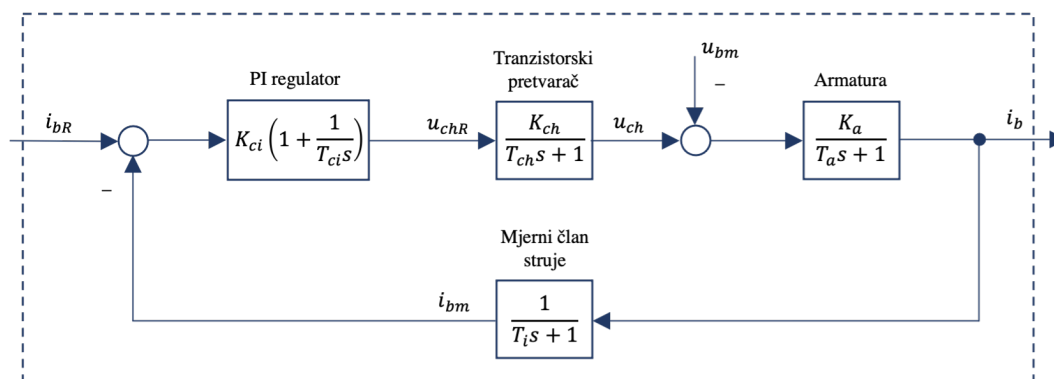
strujom prima referentnu struju i_{bR} . Ta struja je suma maksimalne struje punjenja I_{max} , koja se koristi tijekom faze konstantne struje punjenja, i negativne strujne komande i_{blim} iz superponiranog napona ograničavajućeg regulatora. Taj regulator se aktivira kada izmjenjeni napon na stezaljkama baterije u_{bs} prelazi graničnu vrijednost napona baterije u_{blim} (vidljivo na slici 39 kao blok mrtve zone).

Referenca [14] pruža detaljnu proceduru za podešavanje naponskog PI regulatora. Sveobuhvatna analiza u toj referenci pokazuje da korištenje ove strategije upravljanja rezultira naponom na stezaljkama baterije koji se asimptotski približava željenoj vrijednosti napona otvorenog kruga $U_{oc}(\xi_R)$, koja je povezana s željenim ciljnim stanjem napunjenosti ξ_R , dok se struja punjenja baterije i_b približava nuli.



Slika 39: Kaskadna regulacija punjenja baterije

Već rečeno, podređeni regulacijski krug ćemo raspisati detaljnije da vidimo točno odakle nam proizlaze vremenske konstante koje su bitne za stabilnost sustava jer je podređeni krug na stabilnost (brži je).



Slika 40: Podređeni regulacijski krug struje

Vrijeme uzorkovanja (T), vrijeme pretvarača (T_{ch}), vrijeme armature (T_a) i vrijeme senzora struje možemo zbrojiti sve pod jednu parazitsku vremensku konstantu jer

vremenske konstante tih vrijednosti su male (ne utječu na stabilnost sustava u tolikoj mjeri).

$$T_{\sum pi} = T + T_{ch} + T_a + T_i \quad (5.2)$$

Gdje je T_a :

$$T_a = \frac{L_c}{R_c} \quad (5.3)$$

S obzirom da je na slici 40 prikazan podređeni regulacijski krug te je on svakako brži od nadređenog, možemo cijeli sustav aproksimirati sa PT_1 član uz uvjet da uzmemo dovoljnu veliku vremensku konstantu koja opisuje taj krug. Integracijski član T_{ci} možemo zanemariti u ovom slučaju jer znamo da njegova vremenska konstanta je dovoljno mala za zanemariti. No, radi sigurnosti i stabilnosti, Nadomjesna vremenska konstanta se računa na sljedeći način:

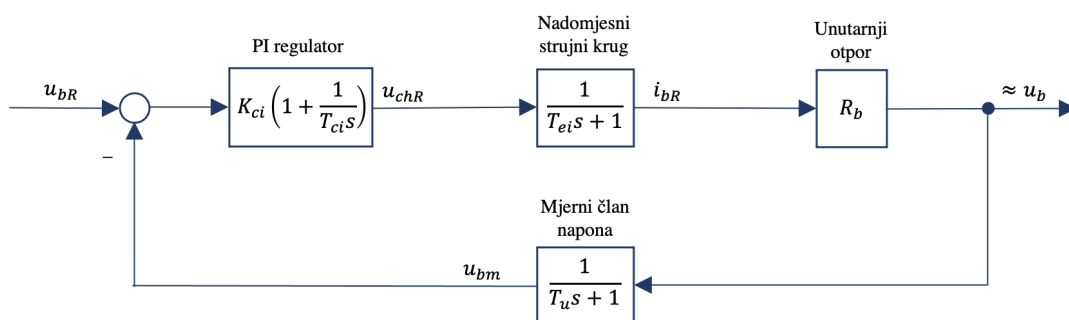
$$T_{ei} = \frac{T_{\sum pi}}{D_2} = 2 \cdot T_{\sum pi} \quad (5.4)$$

S time da je član $D_2 = 0.5$.

Stoga, PT_1 član izgleda:

$$G_i(s) = \frac{1}{T_{ei}s + 1} \quad (5.5)$$

S postavljenim uvjetima, regulator i sustav upravljani po naponu za izračun vrijednosti regulatora izgleda:



Slika 41: Nadređeni regulacijski krug napona

Koristeći se blokovskom algebrom dobivamo izvod:

$$G_1 = \frac{K_{cu}(T_{cu}s + 1)}{T_{cu}s}$$

$$G_2 = \frac{1}{T_{ei}s + 1}$$

$$G_3 = R_b$$

$$G_4 = \frac{1}{T_us + 1}$$

$$G_{234} = \frac{1}{T_{ei}s + 1} \cdot R_b \cdot \frac{1}{T_us + 1}$$

$$G_{234} = \frac{R_b}{(T_{ei}T_u)s^2 + (T_{ei} + T_u)s + 1}$$

$$G_{1234} = \frac{K_{cu}(T_{cu}s + 1)}{T_{cu}s} \cdot \frac{R_b}{(T_{ei}T_u)s^2 + (T_{ei} + T_u)s + 1}$$

$$G_{1234} = \frac{K_{cu}R_b(T_{cu}s + 1)}{(T_{ei}T_uT_{cu})s^3 + (T_{cu}(T_{ei} + T_u))s^2 + T_{cu}s + 1}$$

$$G_O = G_{1234} = \frac{T_{ei}T_uT_{cu}}{K_{cu}R_b}s^3 + \frac{T_{cu}(T_{ei} + T_u)}{K_{cu}R_b}s^2 + \frac{T_{cu}}{K_{cu}R_b}s$$

Za Izračun prijenosne funkcije $G(s) = \frac{u_b}{u_{bR}}$ poslužit ćemo se sljedećom formulom:

$$G(s) = \frac{G_O}{1 + G_O} \quad (5.6)$$

Te dobivamo sljedeći zapis:

$$G(s) = \frac{\frac{T_{cu}s + 1}{\frac{T_{ei}T_uT_{cu}}{K_{cu}R_b}s^3 + \frac{T_{cu}(T_{ei} + T_u)}{K_{cu}R_b}s^2 + \frac{T_{cu}}{K_{cu}R_b}s}}{1 + \frac{T_{cu}s + 1}{\frac{T_{ei}T_uT_{cu}}{K_{cu}R_b}s^3 + \frac{T_{cu}(T_{ei} + T_u)}{K_{cu}R_b}s^2 + \frac{T_{cu}}{K_{cu}R_b}s}}$$

Uz sređivanje izraza dobivamo:

$$G(s) = \frac{T_{cu}s + 1}{\frac{T_{ei}T_uT_{cu}}{K_{cu}R_b}s^3 + \frac{T_{cu}(T_{ei} + T_u)}{K_{cu}R_b}s^2 + \left(\frac{T_{cu}}{K_{cu}R_b} + T_{cu}\right)s + 1} \quad (5.7)$$

Karakteristična jednadžba Prijenosne funkcije sustava glasi:

$$A = \frac{T_{ei}T_uT_{cu}}{K_{cu}R_b}s^3 + \frac{T_{cu}(T_{ei} + T_u)}{K_{cu}R_b}s^2 + \left(\frac{T_{cu}}{K_{cu}R_b} + T_{cu} \right) s + 1 \quad (5.8)$$

Ako ju izjednačimo sa jednadžbom (5.1) dobivamo sljedeće izraze:

$$\begin{aligned} \frac{T_{ei}T_uT_{cu}}{K_{cu}R_b} &= D_3D_2^2T_{eu}^3 \\ \frac{T_{cu}(T_{ei} + T_u)}{K_{cu}R_b} &= D_2T_{eu}^2 \\ \frac{T_{cu}}{K_{cu}R_b} + T_{cu} &= T_{eu} \end{aligned} \quad (5.9)$$

Raspisom prve jednadžbe u (5.9) dobivamo sljedeće:

$$T_{cu} = \frac{D_3D_2^2T_{eu}^3K_{cu}R_b}{T_{ei}T_u} \quad (5.10)$$

Uz sređivanje druge jednadžbe u (5.9) dobivamo sljedeće:

$$T_{eu} = \frac{1}{D_3D_2} \cdot \frac{T_{ei}T_u}{T_{ei} + T_u} \quad (5.11)$$

Uz pomoć jednadžbi (5.10) i (5.11) možemo izračunati izraz pojačanja reguatora:

$$\begin{aligned} T_{cu} \left(\frac{1}{K_{cu}R_b} + 1 \right) &= T_{eu} \\ \frac{D_3D_2^2T_{eu}^3K_{cu}R_b}{T_{ei}T_u} \left(\frac{1}{K_{cu}R_b} + 1 \right) &= T_{eu}^2 \\ K_{cu}R_b + 1 &= \frac{1}{D_3D_2^2} \cdot \frac{T_{ei}T_u}{T_{eu}} \end{aligned}$$

Te pojačanje regulatora glasi:

$$K_{cu} = \frac{1}{D_3D_2^2} \cdot \frac{T_{ei}T_u}{T_{eu}} - \frac{1}{R_b} \quad (5.12)$$

5.3. Upravljanje temeljeno na neizrazitoj logici

Neizrazita logika, također poznata kao logika neodređenosti ili logika granica, je matematički formalizam koji omogućuje rukovanje neodređenim ili nepreciznim vrijednostima. Dok se klasična binarna logika temelji na istinitosnim vrijednostima "istina" (1) ili "laž" (0), neizrazita logika uvodi kontinuirani spektar vrijednosti između potpune istine i potpune laži. Ova logika koristi koncept neizrazitih skupova i pravila zaključivanja kako bi se nosila s nesigurnostima i neodređenostima u stvarnim svjetskim situacijama. Ključni elementi neizrazite logike su neizraziti skupovi i neizrazite funkcije pripadnosti. Neizraziti skupovi omogućuju opisivanje vjerojatnosti ili stupnja pripadnosti elementa određenom skupu kroz funkciju pripadnosti. Ova funkcija dodjeljuje vrijednost između 0 i 1 kako bi se odredio stupanj pripadnosti. Na primjer, u kontekstu upravljanja automatskim klimom uređajem, možemo imati neizraziti skup "toplo" s funkcijom pripadnosti koja određuje koliko je nešto "toplo" na skali od 0 do 1.

Upravljanje temeljeno na neizrazitoj logici koristi se za donošenje odluka i upravljanje sustavima koji uključuju nesigurnost ili imaju mnogo neodređenih ulaznih vrijednosti. Umjesto binarnih logičkih pravila, neizrazita logika koristi pravila zaključivanja koja se temelje na neizrazitim vrijednostima. Upravljački sustavi koji koriste neizrazitu logiku mogu biti fleksibilniji i učinkovitiji u rješavanju problema u stvarnom svijetu, gdje je teško ili nemoguće dobiti precizne ulazne podatke. Upravo takvi regulatori pokrivaju nelinearne sustave gdje linearizacija modela jednostavno nije moguća.

Upravljanje temeljeno na neizrazitoj logici obično uključuje sljedeće korake:

- Fazifikacija
- Neizrazito odlučivanje
- Defazifikacija

5.3.1. Fazifikacija

Fazifikacija je proces transformiranja ulaznih izrazitih vrijednosti u stupnjeve pripadnosti ulaznih neizrazitih skupova. Glavni dio postupka je:

- Određivanje ulazno/izlaznih varijabli
- Određivanje područja definicije navedenih varijabli

- Određivanje broja i oblika funkcija pripadnosti koje prekrivaju područje definicije pojedinih jezičnih varijabli

Za naš model smo uzeli dvije ulazne jezične varijable, estimirana vrijednost stanja naponjenosti baterije te napon baterije, a za izlaznu izrazitu varijablu smo uzeli pretvarač napona.

Ulazne vrijednosti:

$$A = (A_1, A_2, \dots, A_n) \rightarrow p_i, \quad B = (B_1, B_2, \dots, B_n) \rightarrow q_i$$

Izlazne vrijednosti:

$$C = (C_1, C_2, \dots, C_n) \rightarrow r_{ij}$$

5.3.2. Neizrazito odlučivanje

Neizrazito odlučivanje je proces transformiranja ulaznih neizrazitih skupova u izlazne neizrazite skupove.

Izlaznu neizrazitu funkciju pripadnosti dobivamo kao agregaciju navedenih pravila primjenom operatora logičkog zbrajanja:

$$\mu_{agg}(x, y, u) = ((p_1 \wedge q_1) \rightarrow r_{11}) \vee \dots \vee ((p_n \wedge q_m) \rightarrow r_{nm}) \quad (5.13)$$

5.3.3. Defazifikacija

Defazifikacija je proces transformiranja neizrazitih izlaznih skupova u izrazite izlazne vrijednosti. Taj proces je nužan kod neizrazitih regulatora s obzirom da izvršni član (aktuator) može raditi samo s izrazitim vrijednostima.

Koristeći se metodom težišta, dobivamo izrazitu izlaznu vrijednost:

$$u_0 = \frac{\int u \cdot \mu_{agg}(x, y, u) du}{\int \mu_{agg}(x, y, u) du} \quad (5.14)$$

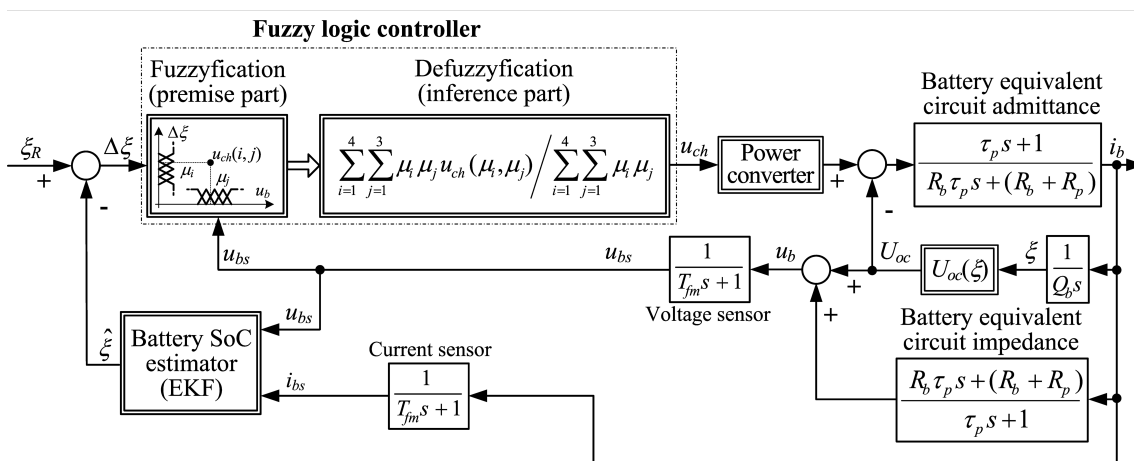
5.4. Dizajn neizrazitog regulatora

Blok dijagram na slici 42 prikazuje sustav upravljanja punjenjem baterije temeljen na neizrazitoj logici koji se sastoji od neizrazitog regulatora s dvije povratne varijable: razlikom SoC baterije $\Delta\xi$ (pogreška upravljanja) i naponom terminala baterije. Lingvistička pravila koja definiraju premisni dio određenog zakona neizrazitog upravljanja stoga su dana u sljedećem obliku za k -ti korak uzorkovanja:

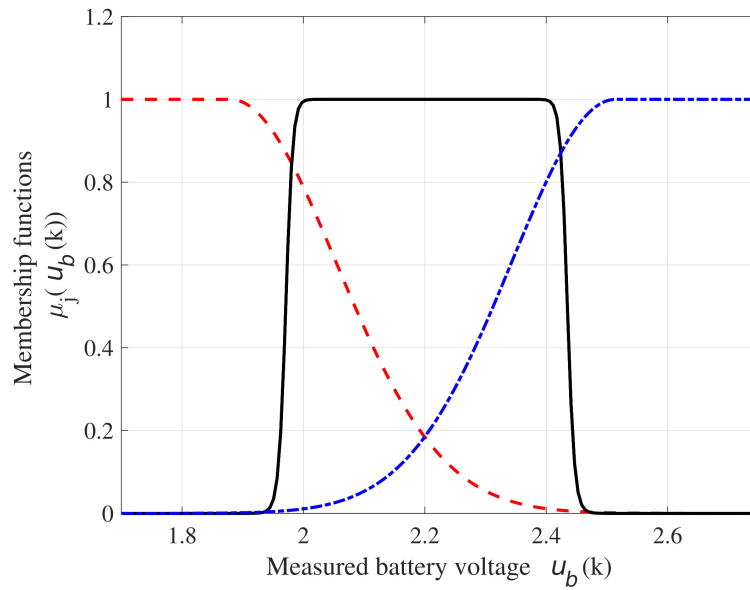
$$\mathbf{IF} \Delta\xi \in \mu_i(\Delta\xi(k)) \ \& \ u_b(k) \in \mu_j(u_b(k)) \ \mathbf{THEN} \ u_{ch}(i, j) = u_{ch}(\Delta\xi(k), u_b(k)) \quad (5.15)$$

Uz funkcije pripadnosti greške upravljanja $\Delta\xi(k)$ i napona na terminalu baterije $u_b(k)$, definiramo ih kao $\mu_i(\Delta\xi(k))$ i $\mu_j(u_b(k))$. U izlaznom dijelu neizrazitog regulatora, funkcija $u_{ch}(\cdot)$ predstavlja izlazne funkcije u inferencijskom dijelu. Taj neizraziti regulator je definiran kao solitoni konstantne vrijednosti, čije vrijednosti odgovaraju željenoj naponskoj naredbi za DC/DC energetske pretvarač koji napaja bateriju. Defazificirani izlaz neizrazitog regulatora, poznat kao "oštri" izlaz, računa se na sljedeći način (vidljivo na slici 8):

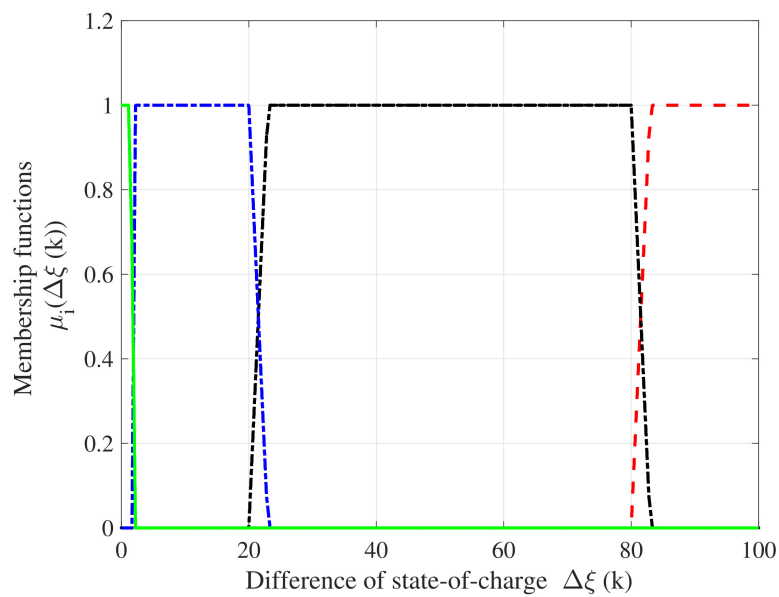
$$u_{ch}(k) = \frac{\sum_{i=1}^4 \sum_{j=1}^3 \mu_i(\Delta\xi(k)) \mu_j(u_b(k)) u_{ch}(\mu_i, \mu_j)}{\sum_{i=1}^4 \sum_{j=1}^3 \mu_i(\Delta\xi(k)) \mu_j(u_b(k))} \quad (5.16)$$



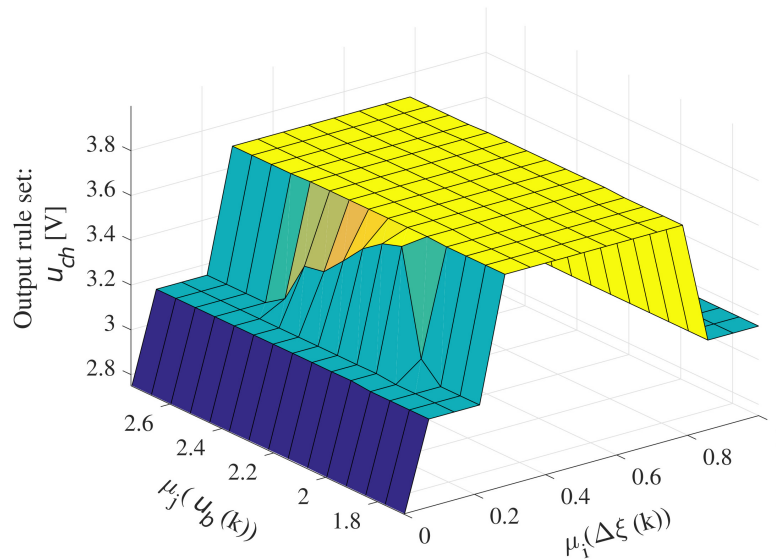
Slika 42: Blok dijagram neizrazitog regulatora kao punjač baterije



Slika 43: Funkcije pripadnosti neizrazitog regulatora s obzirom na signal pogreške upravljanja SoC-a



Slika 44: Funkcije pripadnosti neizrazitog regulatora s obzirom na signal napona baterije



Slika 45: Vrijednosti izlaznog pravila solitona

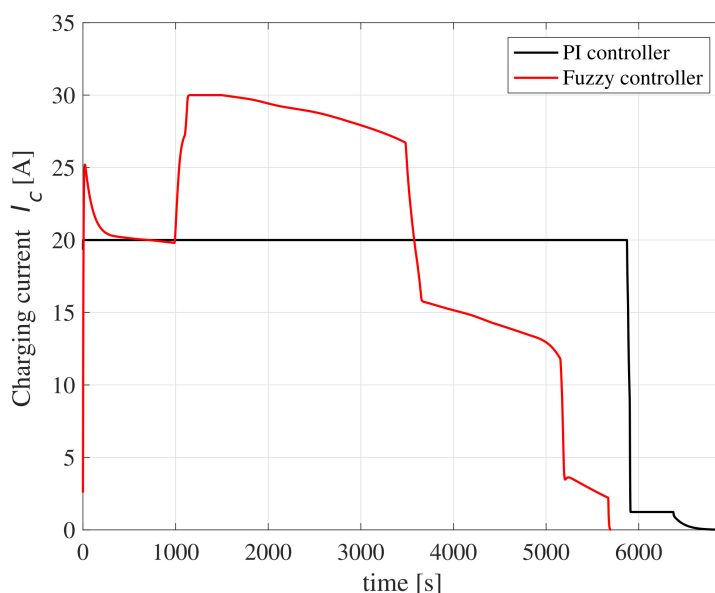
Na slikama 43, 44 i 45 su prikazane funkcije pripadnosti $\mu_i(\Delta\xi(k))$ i $\mu_j(u_b(k))$ te vrijednosti izlazne funkcije u_{ch} u izlaznom skupu predloženog regulatora temeljenog na neizrazitoj logici. Za tu svrhu koriste se izgladeni trapezoidni oblici funkcija pripadnosti, kako je prikazano na slikama 43 i 44. Te funkcije pripadnosti odgovaraju četirima karakterističnim SoC (stanje napunjenosti baterije) režimima (slika 44): vrlo niski SoC (zeleni puni krivulja), niski do srednji SoC (plavi isprekidani krivulja), srednji do visoki SoC (crna crtica isprekidani krivulja) i visoki SoC (crveni isprekidani krivulja). Slično, ulazni skup za napon na priključku baterije u_b je podijeljen na niski do srednji napon (crvena isprekidana krivulja), srednji napon (crna puna krivulja) i visoki napon (plava isprekidana krivulja), kako je prikazano na slici 43. Funkcionalne (solitonske) vrijednosti u izlaznom dijelu zakona neizrazitog upravljanja (slika 45) dodjeljuju veću važnost SoC vrijednostima u srednjem do visokom rasponu, što je također prilagođeno s obzirom na napon terminala baterije kako bi se riješili mogući prekoračaji struje baterije.

5.5. Usporedba između neizrazite logike i konvencionalnog sustava upravljanja punjenjem baterije

Oba prikazana sustava kontrole punjenja baterija su pažljivo verificirana kroz simulacije na nelinearnom modelu LTO (Litij-ionski titanat) baterijske ćelije koji je implementiran u okruženju Matlab/Simulink. Korištenjem ovog simulacijskog okruženja omogućena je temeljita analiza i evaluacija performansi sustava.

Posebna pažnja posvećena je učinkovitosti SoC estimatora koji se temelji na EKF-u te u kombinaciji sa regulatorom koji je temeljen na neizrazitoj logici. Kroz simulacije je potvrđeno da SoC estimator pruža pouzdane rezultate u procjeni stanja napunjenosti baterije.

Slika 46 pokazuje struje punjenja neizrazitog regulatora te PI regulatora. Vidimo da neizraziti regulator ima drugačije ponašanje, točnije pulsno. Njegova izvedba je napravljena na takav način da kada regulator prepozna da nismo više u području visokog otpora(najmanje će se disipirati toplina) da može puniti sa maksimalnom strujom od 30 A. Također se vidi da je zadnja faza punjenja pri niskim vrijednostima iz razloga da bateriju postepeno nadopunjavamo do 100 %. Sa istim srednjim strujama, dobivamo da je Neizraziti regulator brži za punih 20 minuta s time da je razlika između PI i neizrazitog regulatora u konačnom naponu 0.02 %.



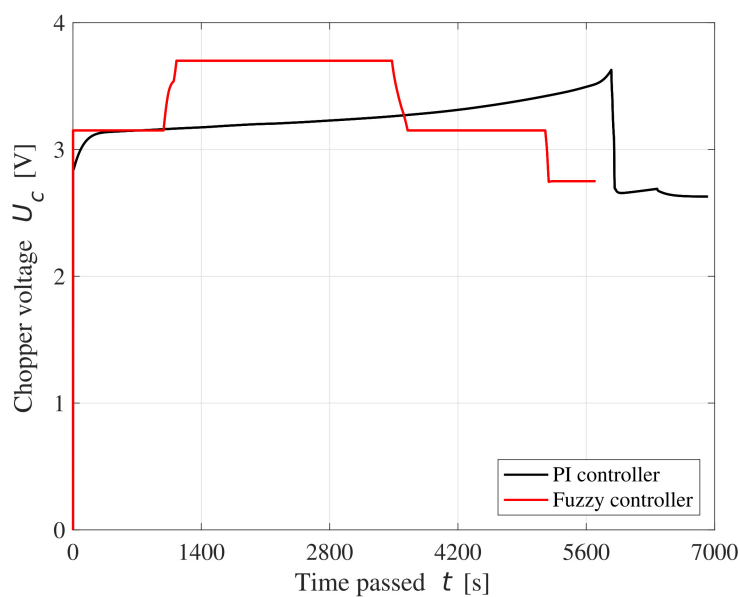
Slika 46: Struja punjenja; PI vs Neizrazitog regulatora

Slika 47 pokazuje Reference napona neizrazitog regulatora koje se koriste za pretvarač sa sljedećim vrijednostima:

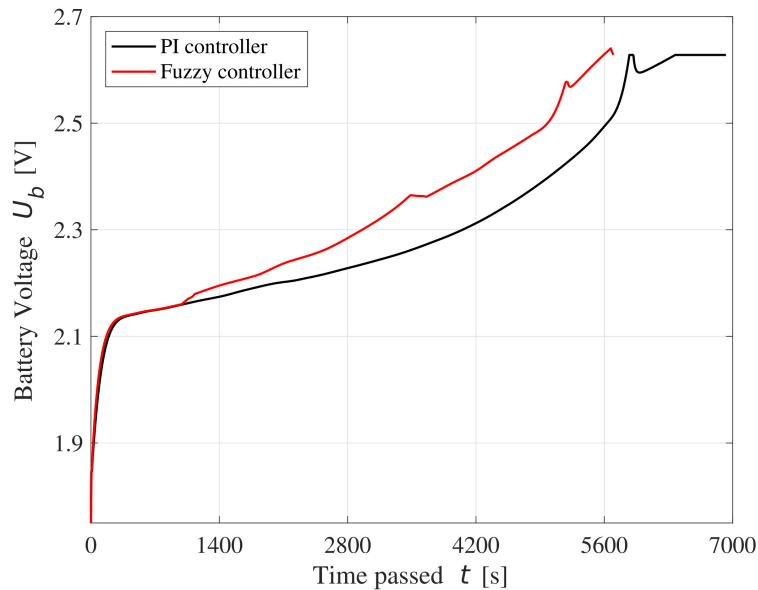
- *Normal charge* - 3.15 V
- *Fast charge* - 3.7 V
- *End charge* - 2.75 V

Naravno, i u jednom i u drugom regulatoru je ta faza *Constant Current*, a tek kraj faze punjenja dolazi do *Constant Voltage* jer težimo prema referentnom u_{bR} naponu.

Dok, PI regulator sukladno prati krivulju napona baterije sa pomakom od 1 V. Pred sami kraj, pada na vrijednost 2.628 V i ostaje na toj vrijednosti.



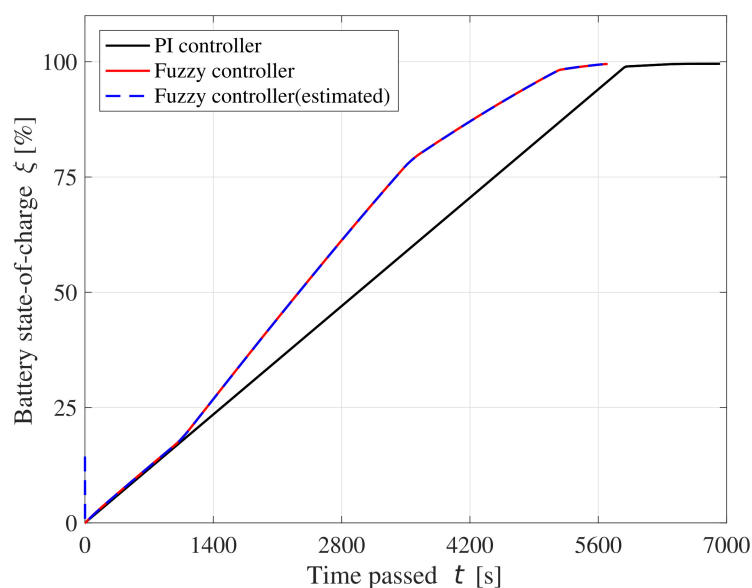
Slika 47: Napon pretvarača; PI vs Neizraziti regulator



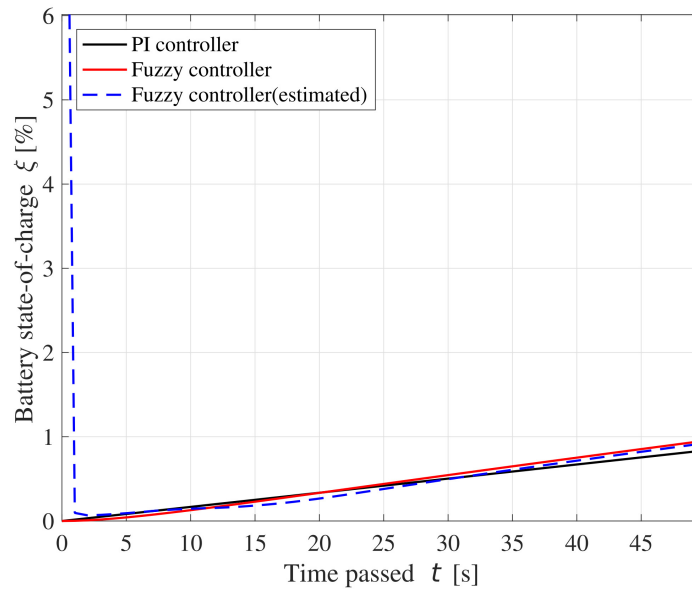
Slika 48: Napon baterije; PI vs Neizraziti regulator

Slika 49 i 50 najbolje prikazuju prednost neizrazitog regulatora u odnosu na PI. Do 20 % punjenje je identično i sve do 95 % se vidi veliki odskok u punjenju. Nadalje, estimirana vrijednost stanja napunjenosti baterije radi dosta precizno i ne predstavlja nikakve smetnje nagle promjene struja kod neizrazitog regulatora.

Kraj punjenja je izvedeno na sljedeći način; Za PI je korišteno da kada je struja I_c manja od 10 mA da punjenje završava, a za neizraziti regulator je korišteno da kada dođe do 99.5 % SoC-a da punjenje završava.



Slika 49: Stanje napunjenosti baterije; PI vs Neizraziti regulator



Slika 50: Uvećani početni dio prethodne slike

Neizrastiti regulator je dizajniran s viskom točnošću te pokazuje odlike u ponašanju sa nelinearnim sustavom. Također, upućuje na jednostavniji matematički pristup u odnosu sa referentnim punjačem (ne zahtijeva računanje pojačanja), no potrebno je eksperimentalno odrediti granice same baterije.

6. Zaključak

U radu su prikazani rezultati eksperimentalne karakterizacije suvremene litij-titanat (LTO) elektrokemijske baterije (30Ah/2,4V/6C) u svrhu modeliranja upravljački orijentiranog modela baterije prikladnog za dizajn estimatora stanja napunjenosti (*SoC*). Identifikacija modela baterijske ćelije provedena je u tri različite faze. Karakteristika kapaciteta punjenja i krivulja napona otvorenog kruga u odnosu na *SoC* najprije su snimljene korištenjem pražnjenja i punjenja konstantnom strujom, pri čemu se u prvom slučaju baterija praznila dok napon na stezaljkama nije dosegao graničnu vrijednost od 1,7 V. Nadalje, intervali punjenja isprepleteni su s intervalima rekuperacije napona baterije u cijelom rasponu napona baterije, a rezultirajuća krivulja napona otvorenog kruga u odnosu na *SoC* je aproksimirana kubičnim *spline*-ovima. Konačno, identifikacija serijskog otpora baterijske ćelije, polarizacijskog otpora i vremenske konstante kašnjenja polarizacije provedena je za različite vrijednosti struje punjenja korištenjem procjene parametara najmanjih kvadrata na temelju testnog signala pseudoslučajnog binarnog niza.

Tako dobiveni model ekvivalentnog strujnog kruga baterije karakteriziraju promjenjivi parametri (ovisni o *SoC*-u i struji), raspoređeni su u jednodimenzionalne (1D) ili dvodimenzionalne (2D) statičke mape, što je pogodno sa stajališta dizajna simulacijskog modela usmjerenog na upravljanje baterije. Dobiveni simulacijski model verificiran je u usporedbi s eksperimentalno snimljenim podacima o bateriji, a rezultati simulacijskog modela pokazali su visoko slaganje s eksperimentima, ukazujući tako na dobru točnost modela baterije, posebno u srednjem *SoC* području ($10\% \leq \text{SoC} \leq 90\%$). Odstupanja modela u graničnim *SoC* regijama ($0\% \leq \text{SoC} \leq 10\%$ i $90\% \leq \text{SoC} \leq 100\%$) nisu naglašena, što ukazuje na dobru ukupnu točnost eksperimentalno identificiranog modela baterije.

Na temelju modela ekvivalentnog strujnog kruga baterije, prošireni Kalmanov filtar (EKF) dizajniran je da služi kao estimator stanja napunjenosti, a njegova učinkovitost je potvrđena pomoću simulacija. Rezultati su pokazali da predloženi estimator stanja temeljen na EKF-u može pratiti stanje napunjenosti baterije s visokom točnošću, a karakterizira ga i brza konvergencija (brzi odgovor) estimiranog *SoC*-a iz neusklađenog početnog stanja.

Modeli PI i neizrazitog regulatora su dizajnirani prema prijašnje napravljenim modelima. Ovi regulatori služe za reguliranje procesa punjenja baterije, svaki temeljen na vlastitoj topologiji sustava punjenja i principu upravljanja strujom i naponom baterije. Rezultati su pokazali da dizajnirani neizraziti regulator funkcionira odlično i nije karakteriziran značajnim odstupanjem *SOC*-a prilikom završavanja punjenja. Također, zbog visoke točnosti EKF-a regulator funkcionira jako dobro na nagle promjene struje, točnije

pri promjeni režima punjenja.

Bibliografija

- [1] Victoria, M., Zhu, K., Brown, T., Andresen, G. B., and Greiner, M., The role of storage technologies throughout the decarbonisation of the sector-coupled European energy system, *Energy Conversion and Management*, Vol. 201, Paper no. 111977, 17 pages, 2019. <https://doi.org/10.1016/j.enconman.2019.111977>.
- [2] Sandhya, C. P., John, B., and Gouri, C., Lithium titanate as anode material for lithium-ion cells: a review, *Ionics*, Vol. 20, pp. 601–620, 2014. <https://doi.org/10.1007/s11581-014-1113-4>.
- [3] Liang, Z., Hui-Lin, P., Yong-Sheng, H., Hong, L., and Li-Quan, C., Spinel lithium titanate ($\text{Li}_4\text{Ti}_5\text{O}_{12}$) as novel anode material for room-temperature sodium-ion battery, *Chinese Physics B*, Vol. 21, No. 2, Paper No. 028201, 5 pages, 2012. <https://doi.org/10.1088/1674-1056/21/2/028201>.
- [4] Zhang, X., Peng, H., Wang, H., and Ouyang, M., Hybrid Lithium Iron Phosphate Battery and Lithium Titanate Battery Systems for Electric Buses, *IEEE Transactions on Vehicular Technology*, Vol. 67, No. 2, pp. 956–965, 2018. <https://doi.org/10.1109/TVT.2017.2749882>.
- [5] Nemeth, T., Schröer, P., Kuipers, M., and Sauer, D. U., Lithium titanate oxide battery cells for high-power automotive applications – Electro-thermal properties, aging behavior and cost considerations, *Journal of Energy Storage*, Vol. 31, paper No. 101656, 14 pages, 2020. <https://doi.org/10.1016/j.est.2020.101656>.
- [6] C. Zhu, X. Li, L. Song, L. Xiang, "Development of a theoretically based thermal model for lithium ion battery pack", *Journal of Power Sources*, 2012.
- [7] Zhai, G., Liu, S., Wang, Z., Zhang, W., and Ma, Z., State of energy estimation of lithium titanate battery for rail transit application, *Energy Procedia*, Vol. 105, pp. 3146–3151, 2017. <https://doi.org/10.1016/j.egypro.2017.03.681>.
- [8] Kvaternik, K., Pavković, D., Kozhushko, Y., and Cipek, M., Extended Kalman Filter Design for State-of-Charge Estimation of a Lithium-Titanate Battery Cell, *Proceedings of the 2022 International Conference on Smart Sys-*

- tems and Technologies (SST), Osijek, Croatia, 2022, pp. 249–254. <https://doi.org/10.1109/SST55530.2022.9954682>.
- [9] Pavković, D., Premec, A., Krznar, M., and Cipek, M., Current and voltage control system designs with EKF-based state-of-charge estimator for the purpose of LiFePO₄ battery cell charging, *Optimization and Engineering*, Vol. 23, No. 4, pp. 2235–2263, 2022. <https://doi.org/10.1007/s11081-022-09728-1>.
- [10] Schröer, P., Khoshbakht, E., Nemeth, T., Kuipers, M., Zappen, M., and Sauer, D. U., Adaptive modeling in the frequency and time domain of high-power lithium titanate oxide cells in battery management systems, *Journal of Energy Storage*, Vol. 32, paper No. 101966, 19 pages, 2020. <https://doi.org/10.1016/j.est.2020.101966>.
- [11] Namor, E., Torregrossa, D., Sossan, F., Cherkaoui, R., and Paolone, M., Assessment of battery ageing and implementation of an ageing aware control strategy for a load leveling application of a lithium titanate battery energy storage system, *Proceedings of 2016 IEEE 17th Workshop on Control and Modeling for Power Electronics (COMPEL)*, Trondheim, Norway, 2016, pp. 1-6. <https://doi.org/10.1109/COMPEL.2016.7556779>.
- [12] Roscher, M. A. and Sauer, D. U., Dynamic electric behavior and open-circuit-voltage modeling of LiFePO₄-based lithium-ion secondary batteries, *Journal of Power Sources*, Vol. 196, No. 1, pp. 331–336, 2011. <https://doi.org/10.1016/j.jpowsour.2010.06.098>.
- [13] Shi, S., Xie, Y., Li, M., Yuan, Y., Yu, J., Wu, H., Liu, B., and Liu, N., Non-steady experimental investigation on an integrated thermal management system for power battery with phase change materials, *Energy Conversion and Management*, Vol. 138, pp. 84–96, 2017. <https://doi.org/10.1016/j.enconman.2017.01.069>.
- [14] Chen, A., Zhang, W., Zhang, C., Huang, W., and Liu, S., A Temperature and Current Rate Adaptive Model for High-Power Lithium-Titanate Batteries Used in Electric Vehicles, *IEEE Transactions on Industrial Electronics*, Vol. 67, No. 11, pp. 9492–9502, 2020. <https://doi.org/10.1109/TIE.2019.2955413>.
- [15] Sun, S., Wang, Q., Zhao, X., Sun, J., and Lin, Z., Numerical study on lithium titanate battery thermal response under adiabatic condition, *Energy Conversion and Management*, Vol. 92, pp. 184–193, 2015. <https://doi.org/10.1016/j.enconman.2014.12.019>.
- [16] ELERIX EX-30TK Extreme Power LTO Cell, VDA size 173/100, Technical Specification (<https://shop.gwl.eu/LTO-technology/Cylindrical-Lithium-Titanate-Oxid-Battery-Cell-LTO-2-3V-30AH.html>) [Accessed on April 3rd, 2023]

- [17] Ceraolo, M., New Dynamical Models of Lead-Acid Batteries, IEEE Transactions on Power Systems, Vol. 15, No. 4, pp. 1184-1190, 2000. <https://doi.org/10.1109/59.898088>.
- [18] Ljung, L., System identification—theory for the user, Prentice Hall, Englewood Cliffs, New Jersey, USA, 1987.
- [19] Grewal, M.S., and Andrews, A.P., Kalman filtering—theory and practice, John Wiley and Sons Ltd., 2001.
- [20] Isermann, R., Digital control systems, Vol. 1, Springer-Verlag, Berlin, Germany, 1989
- [21] Cong Zhu, Xinghu Li, Lingjun Song*, Liming Xiang. Development of a theoretically based thermal model for lithium ion battery pack, Elsevier, 2012.

Prilog

Dodatak A

A.1 - *Main* skripta

```
%Diplomski rad, Karlo Kvaternik

%-----%
clear all
clc
%-----%

%Dodavanje foldera s metodama i mjerenim podacima
%-----
addpath(genpath('Methods'))
addpath(genpath('LTO_map'))

%Ucitavanje mapa modela i mjerenih podataka sa baterije
%-----
load('modelValuesDis.mat')
load('modelValuesChg.mat')
load('measuredDataDis.mat')
load('measuredDataChg.mat')
load('ltoMap.mat')

%Vrijednosti modela
%-----
QDis = 31.857; %mean value of all discharged capacities
QChg = 33.0873; %mean value of all charged capacities
```

```

%----- inital values - Discharge - Experimental data
-----%
Qmax24Dis = 31.2104;
Qmax18Dis = 31.5904;
Qmax12Dis = 31.9704;
Qmax6Dis = 32.6475;
%----- inital values - Charge - Experimental data
-----%
Qmax24Chg = 32.9569;
Qmax18Chg = 33.0159;
Qmax12Chg = 33.1286;
Qmax6Chg = 33.2477;

QmaxChg = [Qmax6Dis; Qmax12Dis; Qmax18Dis; Qmax24Dis];
xib = linspace(0,24,1001);
QMAX=interp1(Imap(5:end),QmaxChg,xib,'pchip');

%Ostale vrijednosti modela
%-----
% Filtar mjerenja
Tfilt=4e-3; %[s]
% Chopper
Tch = 1e-3; % [s]
Kch = 1.0; % [-]
% Parametri zavojnice
Rc = 0.05; % [Ohm]
Lc = 0.7e-3;
Tc = Lc/Rc;
% Vrijeme uzorkovanja
T = 0.1; % 10 milisek.
% smetnje
R = 1.0e-3;
%Trajanje simulacije
T_pi = 10000;
%Data acquistion time
Tacq = 1;

%Promjenjivi parametri za estimator
%-----%

```

```

% Varijance perturbacija u stanjima
qk11 = 1e-6; %qj11 = 1e-5, qk22=1e-6 rk = 10
qk22 = 1e-6;
% qk22 = 1.0e2;
Qk = [qk11 0; 0 qk22];
rk = 1e1;
P0 = 1e-1*eye(2,2);
%-----%

% Struja praznjenja u sekvenci
Idis = -15;
% Struja punjenja u sekvenci
Ich = 5;
% Filtar (dinamika) struje baterije od pretvara?a
Tbat =2; % [s]

%% PI regulator
T = 0.01; % 10 milisek.
Tsig0 = Tch + Tc + Tfilt + T;
D2i = 0.5; D3i = 0.5;
Rb = mean(Rsmap(8, :));
Rtot = Rb + Rc;
Imax = 24;
% Pocetni napon
uc0 = 0.0; % [V]
% Pocetna struja
I0 = 0.0;
% Stacionarni napon DC me?ukruga
Udc0=4; %[V]

%Upravljanje po struja baterije

Tei = 1/(D2i*D3i)*((Tsig0*Lc)/(Tsig0*Rtot+Lc));
Kci = 1/(D2i^2*D3i)*(Tsig0*Lc)/(Tei^2)-Rtot;
Tcu = (D3i*D2i^2*Tei^3*Kci)/(Tsig0*Lc);

```

```

%Upravljanje po naponu baterije
% Tsig1 = 2*Tsig0;
%
% Teu = 1/(D2i*D3i)*((Tsig1*Tfilt)/(Tsig1*Tfilt));
% Kcu = 1/(D2i^2*D3i)*(Tsig1*Tfilt)/(Teu^2*Rb)-1/Rb;
% Tcu = (D3i*D2i^2*Teu^3*Kcu*Rb)/(Tsig1*Tfilt);
%Kompenzator od prenapona baterije
D2c = 0.5;
Tsigc = Tei + T + Tfilt;
Tec = 0.9*Tsigc/D2c;

KR_cmp = (1/Rb)*(Tsigc/D2c/Tec - 1);
TI_cmp = Tec*(1 - D2c*Tec/Tsigc);
KI_cmp = KR_cmp/TI_cmp;

%% Iscrtavanje simulacijskog modela i stvarnog %%
%-----
%% Uoc curve
figure(2)
plot(xk*100,Uoc,'k-','LineWidth',2)
hold on
plot(dxk*100, dUoc,'ro','MarkerSize',8)
xlabel('Battery state-of-charge \xi(k)'),ylabel('Open
circuit voltage {\it U}_{oc} [V]')
xlim([0 100])
ylim([1.7 2.7])
grid on
yticks(1.7:0.2:2.7);
ax = gca;
ax.FontSize = 16;
set(gca,'fontname','times')
legend('Spline interpolation', 'Measurements','Location
','northwest')

%% Discharging curves
figure(1)
plot(Q6dis, ub6dis, 'k','LineWidth',1.5)

```

```

hold on
plot (Q12dis, ub12Dis, 'g', 'LineWidth',1.5)
hold on
plot (Q18dis, ub(4:end), 'r', 'LineWidth',1.5)
hold on
plot (Q24dis, ub24Dis, 'b', 'LineWidth',1.5)
xlabel('Discharged charge {\it Q}_{max} ( {\it I}_d ) [Ah
    ]'),ylabel('Battery terminal voltage {\it U}_b [V]')
xlim([0 33])
ylim([1.7 2.7])
grid on
yticks(1.7:0.2:2.7);
ax = gca;
ax.FontSize = 16;
set(gca, 'fontname', 'times')
legend('6 A', '12 A', '18 A', '24 A', 'Location', '
    northeast')
title(legend, ['Discharging' newline 'current | I_d |'])

%% Rp Rs Tp map figure
figure(2)
plot (DxiMap*100, Rpmap(5,:) *1000, 'b-', 'LineWidth',1.25,
    'MarkerSize',6)
hold on
plot (DxiMap*100, Rpmap(6,:) *1000, 'g-', 'LineWidth',1.25,
    'MarkerSize',6)
hold on
plot (DxiMap*100, Rpmap(7,:) *1000, 'k-', 'LineWidth',1.25, '
    MarkerSize',6)
hold on
plot (DxiMap*100, Rpmap(8,:) *1000, 'r-', 'LineWidth',1.25, '
    MarkerSize',6)
xlabel('Battery state-of-charge \xi [%]'),ylabel(' {\it
    R}_p [m\Omega]')
xlim([0 100])
grid on
xticks(0:20:100);
ax = gca;
ax.FontSize = 16;

```

```

set(gca, 'fontname', 'times')
legend('6 A', '12 A', '18 A', '24 A', 'Location', '
    northeast')
title(legend, ['Charging' newline 'current | I_c |'])

figure(3)
plot(DxiMap*100, Rpmap(4, :)*1000, 'b-', 'LineWidth', 1.25,
    'MarkerSize', 6)
hold on
plot(DxiMap*100, Rpmap(3, :)*1000, 'g-', 'LineWidth', 1.25,
    'MarkerSize', 6)
hold on
plot(DxiMap*100, Rpmap(2, :)*1000, 'k-', 'LineWidth', 1.25, '
    MarkerSize', 6)
hold on
plot(DxiMap*100, Rpmap(1, :)*1000, 'r-', 'LineWidth', 1.25, '
    MarkerSize', 6)
xlabel('Battery state-of-charge \xi [%]'), ylabel('{\it
    R_p} [m\Omega]')
xlim([0 100])
grid on
xticks(0:20:100);
ax = gca;
ax.FontSize = 16;
set(gca, 'fontname', 'times')
legend('6 A', '12 A', '18 A', '24 A', 'Location', '
    northeast')
title(legend, ['Discharging' newline 'current | I_d |'])

figure(4)
plot(DxiMap*100, Rsmmap(4, :)*1000, 'b-', 'LineWidth', 1.25,
    'MarkerSize', 6)
hold on
plot(DxiMap*100, Rsmmap(3, :)*1000, 'g-', 'LineWidth', 1.25,
    'MarkerSize', 6)
hold on
plot(DxiMap*100, Rsmmap(2, :)*1000, 'k-', 'LineWidth', 1.25, '
    MarkerSize', 6)

```



```

hold on
plot(DxiMap*100, Rsmmap(1, :)*1000, 'r-', 'LineWidth', 1.25, '
    MarkerSize', 6)
xlabel('Battery state-of-charge \xi [%]'), ylabel(' {\it
    R_s} [m\Omega]')
xlim([0 100])
grid on
xticks(0:20:100);
ax = gca;
ax.FontSize = 16;
set(gca, 'fontname', 'times')
legend('6 A', '12 A', '18 A', '24 A', 'Location', '
    northeast')
title(legend, ['Discharging' newline 'current | I_d |'])

figure(5)
plot(DxiMap*100, Rsmmap(5, :)*1000, 'b-', 'LineWidth', 1.25,
    'MarkerSize', 6)
hold on
plot(DxiMap*100, Rsmmap(6, :)*1000, 'g-', 'LineWidth', 1.25,
    'MarkerSize', 6)
hold on
plot(DxiMap*100, Rsmmap(7, :)*1000, 'k-', 'LineWidth', 1.25, '
    MarkerSize', 6)
hold on
plot(DxiMap*100, Rsmmap(8, :)*1000, 'r-', 'LineWidth', 1.25, '
    MarkerSize', 6)
xlabel('Battery state-of-charge \xi [%]'), ylabel(' {\it
    R_s} [m\Omega]')
xlim([0 100])
grid on
xticks(0:20:100);
ax = gca;
ax.FontSize = 16;
set(gca, 'fontname', 'times')
legend('6 A', '12 A', '18 A', '24 A', 'Location', '
    northeast')
title(legend, ['Charging' newline 'current | I_c |'])

```

```

figure(6)
plot(DxiMap*100, Rpmap(5,:) .* Cpmap(5,:), 'b-', 'LineWidth'
     ',1.25, 'MarkerSize', 6)
hold on
plot(DxiMap*100, Rpmap(6,:) .* Cpmap(6,:), 'g-', 'LineWidth'
     ',1.25, 'MarkerSize', 6)
hold on
plot(DxiMap*100, Rpmap(7,:) .* Cpmap(7,:), 'k-', 'LineWidth'
     ',1.25, 'MarkerSize', 6)
hold on
plot(DxiMap*100, Rpmap(8,:) .* Cpmap(8,:), 'r-', 'LineWidth'
     ',1.25, 'MarkerSize', 6)
xlabel('Battery state-of-charge \xi [%]'), ylabel(' {\it
      T_p} [s]')
xlim([0 100])
grid on
xticks(0:20:100);
ax = gca;
ax.FontSize = 16;
set(gca, 'fontname', 'times')
legend('6 A', '12 A', '18 A', '24 A', 'Location', '
      northeast')
title(legend, ['Charging' newline 'current | I_c |'])

figure(7)
plot(DxiMap*100, Rpmap(4,:) .* Cpmap(4,:), 'b-', 'LineWidth'
     ',1.25, 'MarkerSize', 6)
hold on
plot(DxiMap*100, Rpmap(3,:) .* Cpmap(3,:), 'g-', 'LineWidth'
     ',1.25, 'MarkerSize', 6)
hold on
plot(DxiMap*100, Rpmap(2,:) .* Cpmap(2,:), 'k-', 'LineWidth'
     ',1.25, 'MarkerSize', 6)
hold on
plot(DxiMap*100, Rpmap(1,:) .* Cpmap(1,:), 'r-', 'LineWidth'
     ',1.25, 'MarkerSize', 6)
xlabel('Battery state-of-charge \xi [%]'), ylabel(' {\it
      T_p} [s]')
xlim([0 100])

```

```

grid on
xticks(0:20:100);
ax = gca;
ax.FontSize = 16;
set(gca, 'fontname', 'times')
legend('6 A', '12 A', '18 A', '24 A', 'Location', '
    northeast')
title(legend, ['Discharging' newline 'current | I_d |'])

```

%% simulation vs model

```

figure(8)
plot(Q6dis, ub6dis, 'b', 'LineWidth', 1.5)
hold on
plot(Q6dis(1:end-2), ub_sim, 'r', 'LineWidth', 1.5)
xlabel('Discharged charge {\it Q}_{max} ( {\it I}_d ) [Ah
    ]'), ylabel('Battery terminal voltage {\it U}_b [V]')
xlim([0 33])
ylim([1.7 2.7])
grid on
yticks(1.7:0.2:2.7);
ax = gca;
ax.FontSize = 16;
set(gca, 'fontname', 'times')
legend('experimental', 'simulation', 'Location', '
    northeast')
title(legend, ['| I_d | = 6 A'])

```

```

figure(9)
plot(Q12dis, ub12Dis, 'b', 'LineWidth', 1.5)
hold on
plot(Q12dis(1:end-1), ub_sim, 'r', 'LineWidth', 1.5)
xlabel('Discharged charge {\it Q}_{max} ( {\it I}_d ) [Ah
    ]'), ylabel('Battery terminal voltage {\it U}_b [V]')
xlim([0 33])
ylim([1.7 2.7])
grid on
yticks(1.7:0.2:2.7);

```

```
ax = gca;
ax.FontSize = 16;
set(gca, 'fontname', 'times')
legend('experimental', 'simulation', 'Location', '
    northeast')
title(legend, ['| I_d | = 12 A'])

figure(10)
plot(Q18dis, ub(4:end), 'b', 'LineWidth', 1.5)
hold on
plot(Q18dis, ub_sim(4:end), 'r', 'LineWidth', 1.5)
xlabel('Discharged charge {\itQ}_{max} ({\it I}_d) [Ah
    ]'), ylabel('Battery terminal voltage {\it U}_b [V]')
xlim([0 33])
ylim([1.7 2.7])
grid on
yticks(1.7:0.2:2.7);
ax = gca;
ax.FontSize = 16;
set(gca, 'fontname', 'times')
legend('experimental', 'simulation', 'Location', '
    northeast')
title(legend, ['| I_d | = 18 A'])

figure(11)
plot(Q24dis, ub24Dis, 'b', 'LineWidth', 1.5)
hold on
plot(Q24dis, ub_sim, 'r', 'LineWidth', 1.5)
xlabel('Discharged charge {\itQ}_{max} ({\it I}_d) [Ah
    ]'), ylabel('Battery terminal voltage {\it U}_b [V]')
xlim([0 33])
ylim([1.7 2.7])
grid on
yticks(1.7:0.2:2.7);
ax = gca;
ax.FontSize = 16;
set(gca, 'fontname', 'times')
legend('experimental', 'simulation', 'Location', '
    northeast')
```

```
        northeast')
title(legend, ['| I_d | = 24 A'])

figure(12)
plot(Q6chg, ub6chg, 'b', 'LineWidth', 1.5)
hold on
plot(Q6chg(1:end-4), ub_sim, 'r', 'LineWidth', 1.5)
xlabel('Charged charge {\it Q}_{max} ({\it I}_d) [Ah]')
    , ylabel('Battery terminal voltage {\it U}_b [V]')
xlim([0 33])
ylim([1.7 2.7])
grid on
yticks(1.7:0.2:2.7);
ax = gca;
ax.FontSize = 16;
set(gca, 'fontname', 'times')
legend('experimental', 'simulation', 'Location', '
        northwest')
title(legend, ['| I_c | = 6 A'])

figure(13)
plot(Q12chg, ub12chg, 'b', 'LineWidth', 1.5)
hold on
plot(Q12chg(1:end-2), ub_sim, 'r', 'LineWidth', 1.5)
xlabel('Charged charge {\it Q}_{max} ({\it I}_d) [Ah]')
    , ylabel('Battery terminal voltage {\it U}_b [V]')
xlim([0 33])
ylim([1.7 2.7])
grid on
yticks(1.7:0.2:2.7);
ax = gca;
ax.FontSize = 16;
set(gca, 'fontname', 'times')
legend('experimental', 'simulation', 'Location', '
        northwest')
title(legend, ['| I_c | = 12 A'])

figure(14)
```

```
plot(Q18chg, ub18chg, 'b', 'LineWidth',1.5)
hold on
plot (Q18chg(1:end-1), ub_sim,'r', 'LineWidth',1.5)
xlabel('Charged charge {\itQ}_{max} ({\it I}_d ) [Ah]')
    ,ylabel('Battery terminal voltage {\it U}_b [V]')
xlim([0 33])
ylim([1.7 2.7])
grid on
yticks(1.7:0.2:2.7);
ax = gca;
ax.FontSize = 16;
set(gca, 'fontname', 'times')
legend('experimental', 'simulation', 'Location', '
    northwest')
title(legend, ['| I_c | = 18 A'])
```

```
figure(15)
plot(Q24chg, ub24, 'b', 'LineWidth',1.5)
hold on
plot (Q24chg, ub_sim,'r', 'LineWidth',1.5)
xlabel('Charged charge {\itQ}_{max} ({\it I}_d ) [Ah]')
    ,ylabel('Battery terminal voltage {\it U}_b [V]')
xlim([0 33])
ylim([1.7 2.7])
grid on
yticks(1.7:0.2:2.7);
ax = gca;
ax.FontSize = 16;
set(gca, 'fontname', 'times')
legend('experimental', 'simulation', 'Location', '
    northwest')
title(legend, ['| I_c | = 24 A'])
```

```
%% SOC - ekf
```

```
figure(17)
plot(t24, SoC_sim*100, 'b', 'LineWidth',1.5)
hold on
```

```

plot (t24, SoC_est*100,'r','LineWidth',1.5)
xlabel('time {\it t} [s]'),ylabel('Battery state-of-
    charge \mu [%]')
xlim([0 5000])
ylim([0 100])
grid on
% yticks(0:20:100);
% xticks(0:1000:5000);

ax = gca;
ax.FontSize = 16;
set(gca,'fontname','times')
legend('Simulation','Estimated','Location','northwest')
title(legend,['| I_c | = 24 A'])

%% current curvature
figure(16)
plot(xib, QMAX,'b-','LineWidth',1.5)
hold on
plot (Imap(5:end), QmaxChg, 'r*','LineWidth',1.5,'
    MarkerSize',6)
xlabel('Discharging current |{\it I}_d| [A]'),ylabel('
    Discharge capacity |{\it Q}_{max}|(|{\it I}_d|) [Ah]')
xlim([ 0 Imap(end)])
ylim([QMAX(end) QMAX(1)])
grid on
ax = gca;
ax.FontSize = 16;
set(gca,'fontname','times')
legend('Interpolation curve','Measurements','Location',
    'northeast')

%% FuzzyVsPi - Controller
clearvars -except ib_fuzzy SoC_Fuzzy ub_Fuzzy SoC_est
    uc_fuzzy t_Fuzzy
save simulation_with_Fuzzy.mat

```

```
plot(t, SoC_sim)

figure(17)
plot(t_PI, SoC_PI*100, 'k-', 'LineWidth', 1.5)
hold on
plot(t_Fuzzy, SoC_Fuzzy*100, 'r-', 'LineWidth', 1.5)
hold on
plot(t_Fuzzy, SoC_est*100, 'b--', 'LineWidth', 1.5, '
    MarkerSize', 6)
xlabel('Time passed {\it t} [s]'), ylabel('Battery
    state-of-charge \mu [%]')
xlim([ 0 6200])
ylim([0 110])
yticks(0:25:100);
xticks(0:1550:6200);
grid on
ax = gca;
ax.FontSize = 16;
set(gca, 'fontname', 'times')
legend('PI controller', 'Fuzzy controller', 'Fuzzy
    controller (estimated)', 'Location', 'northwest')

plot(t_PI, ib_PI)

figure(18)
plot(t_PI, ub_PI, 'k-', 'LineWidth', 1.5)
hold on
plot(t_Fuzzy, ub_Fuzzy, 'r-', 'LineWidth', 1.5)
hold on
xlabel('Time passed {\it t} [s]'), ylabel('Battery
    Voltage {\it U_b} [V]')
xlim([ 0 6200])
ylim([1.75 2.75])
yticks(1.75:0.2:2.75);
xticks(0:1550:6200);
grid on
ax = gca;
ax.FontSize = 16;
```



```

set(gca, 'fontname', 'times')
legend('PI controller', 'Fuzzy controller', 'Location', '
    northwest')

%input one
[xOut, umfOut] = plotmf(fis, 'input', 1);
plot(xOut*100, umfOut(:,1), 'r--', 'LineWidth', 2)
hold on
plot(xOut*100, umfOut(:,2), 'k-.', 'LineWidth', 2)
hold on
plot(xOut*100, umfOut(:,3), 'b-.', 'LineWidth', 2)
hold on
plot(xOut*100, umfOut(:,4), 'g-', 'LineWidth', 2)
xlabel('Difference of state-of-charge \Delta\xi (k)'),
    ylabel(['Membership functions' newline '\mu_i(\
    \Delta\xi (k))'])
grid on
xlim([0 100])
ylim([0 1.2])
xticks(0:20:100);
ax = gca;
ax.FontSize = 16;
set(gca, 'fontname', 'times')

%input two
[xOut, umfOut] = plotmf(fis, 'input', 2);
plot(xOut, umfOut(:,1), 'r--', 'LineWidth', 2)
hold on
plot(xOut, umfOut(:,2), 'k-', 'LineWidth', 2)
hold on
plot(xOut, umfOut(:,3), 'b-.', 'LineWidth', 2)
xlabel('Measured battery voltage {\it u_b} (k)'), ylabel
    (['Membership functions' newline '\mu_j({\it u_b} (
    k))'])
grid on
xlim([1.7 2.75])
ylim([0 1.2])
xticks(1.6:0.2:3.8);
ax = gca;

```

```
ax.FontSize = 16;
set(gca, 'fontname', 'times')

%output
gensurf(fis)
xlabel('\mu_i(\Delta\xi (k))'), ylabel(['\mu_j(\it u_b}
      (k))']), zlabel(['Output rule set:' newline '\it u_
      {ch}} [V]'])
set(get(gca, 'xlabel'), 'rotation', 15)
set(get(gca, 'ylabel'), 'rotation', -30)
zlim([2.75 4])
ax = gca;
ax.FontSize = 16;
set(gca, 'fontname', 'times')
```

A.2 - Skripta za izradu 2D mapa

```
addpath(genpath('Methods'))
addpath(genpath('measuredData'))

%Parametri za izradu mapa
%Charge
%-----
x6 = linspace(0,1,43);
x12 = linspace(0,1,31);
x18 = linspace(0,1,22);
x24 = linspace(0,1,12);

%Discharge
%-----
x6Dis = linspace(1,0,32);
x12Dis = linspace(1,0,24);
x18Dis = linspace(1, 0,13);
x24Dis = linspace(1, 0, 12);

xk = 0:0.001:1;
xkDis = 1:-0.001:0;

%Prvi set mjerenja
%dx_i = [0 0.02579 0.05493 0.07692 0.099374 0.12821
0.15385 0.23077 0.30769 0.38461 0.46154 0.53846
0.61539 0.64103 0.66666 0.69231 0.71795 0.74359
0.76923 0.79487 0.82051 0.84616 0.87180 0.89744
0.92308 0.94872 0.97436 1];
%dUoc = [1.663190 2.068240 2.090715 2.100920 2.113350
2.122406 2.131622 2.153261 2.152316 2.173648
2.190853 2.212600 2.238120 2.249200 2.257410
2.272650 2.287980 2.305020 2.323950 2.336490
2.361500 2.385900 2.406090 2.438520 2.483400
2.519110 2.642650 2.755360];
%2.06987 2.10405 2.09326 2.11610 2.03987 10675
%Drugi set mjerenja
```

```
dx_i = [0 0.02941 0.05882 0.08824 0.11765 0.14706
        0.17647 0.20588 0.29412 0.38235 0.47059 0.55882
        0.64706 0.73529 0.76470 0.79412 0.82353 0.85294
        0.88235 0.91177 0.94118 0.97059 1];
dUoc = [1.75322 2.03987 2.09326 2.10675 2.11610
        2.12451 2.13360 2.13860 2.16270 2.18163 2.19778
        2.22238 2.24607 2.29094 2.30815 2.32132 2.34510
        2.36715 2.39125 2.41715 2.44550 2.48900 2.68371];

%Set mjerenja za temperaturu
dxiT = [0 0.04 0.08 0.16 0.20 0.24 0.28 0.32 0.36 0.40
        0.44 0.48 0.52 0.56 0.60 0.64 0.68 0.72 0.76 0.8
        0.84 0.88 0.92 0.96 1];
dT = [ 17 18 18.8 19.2 19.5 19.6 19.7 20.4 20.6 20.8 21
       21.2 21.3 21.3 21.35 21.36 21.4 21.45 21.48 21.6
       21.7 22 22.4 22.8 23];

%Dobivanje Uoc krivulje

[xi, Uoc] = seg3(dx_i, dUoc);

plot(xk, Uoc, 'r-')
hold on
plot(xk, Uocmap(6, :))
hold on
plot(dx_i, dUoc, 'r.')

[xi, Uoc] = seg3(dx_i, dUoc);

plot(xi, Uoc)
hold on
plot(dx_i, dUoc, 'r.')

%Calculating Uoc gradient for charge and discharge
dUocdSoCChg = gradient(Uoc) ./ (gradient(xk));
plot(xk, dUocdSoCChg)

dUocdSoCDis = gradient(Uocmap(1, :)) ./ (gradient(xk));
```

```
plot(xk, dUocdSoCDis)

dUocdSoCmap = [dUocdSoCDis; dUocdSoCDis; dUocdSoCDis;
               dUocdSoCDis; dUocdSoCChg; dUocdSoCChg; dUocdSoCChg;
               dUocdSoCChg]

%% Mean difference between real model and simulated -
Discharge
%
-----

delta = mean(ub - ub_sim);
delta24 = 7.2; %mV
delta18 = 27.2; %mV
delta12 = 20.8; %mV
delta6 = 25.3; %mV

Delta = [delta6 delta12 delta18 delta24];
avgDelta = mean(Delta)/1000; %V

%Voltage correction for Charge Uoc
Uoc = Uoc - avgDelta;
UocDis = Uoc;

%% Mean difference between real model and simulated -
Charge
%
-----

delta = mean(ub24-ub_sim)
delta24 = 9.7; %mV
delta18 = 9.3; %mV
delta12 = 10.2; %mV
delta6 = 1; %mV

Delta = [delta6 delta12 delta18 delta24];
avgDelta = mean(Delta)/1000; %V

%Voltage correction for Charge Uoc
```

```
UocChg = zeros(1,length(Uoc));
UocChg = Uoc;
%% Charge map
%-----

%6A Parameter identification%
[Rs6,Rp6,Tp6, DQ6]=dynMapChg6() ;

%12A Parameter identification%
[Rs12,Rp12,Tp12, DQ12]=dynMapChg12() ;

%18A Parameter identification%
[Rs18,Rp18,Tp18, DQ18]=dynMapChg18() ;

%24A Parameter identification%
[Rs24,Rp24,Tp24, DQ24]=dynMapChg24() ;

%% Discharge map
%-----

%6A Parameter identification%
[RsDis6,RpDis6,TpDis6, DQDis6]=dynMapDis6() ;

%12A Parameter identification%
[RsDis12,RpDis12,TpDis12, DQDis12]=dynMapDis12() ;

%18A Parameter identification%
[RsDis18,RpDis18,TpDis18, DQDis18]=dynMapDis18() ;

%24A Parameter identification%
[RsDis24,RpDis24,TpDis24, DQDis24]=dynMapDis24() ;

%Identifying virtual capacity
Cp6 = Tp6./Rp6;
Cp12 = Tp12./Rp12;
Cp18 = Tp18./Rp18;
Cp24 = Tp24./Rp24;

CpDis6 = TpDis6./RpDis6;
```

```

CpDis12 = TpDis12./RpDis12;
CpDis18 = TpDis18./RpDis18;
CpDis24 = TpDis24./RpDis24;

%% Discharge
%Rs map
Rs6mapDis=interp1(x6Dis,RsDis6,xkDis,'pchip');
Rs12mapDis=interp1(x12Dis,RsDis12,xkDis,'pchip');
Rs18mapDis=interp1(x18Dis,RsDis18,xkDis,'pchip');
Rs24mapDis=interp1(x24Dis,RsDis24,xkDis,'pchip');
RsMapDis = [Rs6mapDis; Rs12mapDis; Rs18mapDis;
             Rs24mapDis];
RsMapDis = [RsMapDis(4,:); RsMapDis(3,:); RsMapDis(2,:);
            ; RsMapDis(1,:)]

%Rp map
Rp6mapDis=interp1(x6Dis,RpDis6,xkDis,'pchip');
Rp12mapDis=interp1(x12Dis,RpDis12,xkDis,'pchip');
Rp18mapDis=interp1(x18Dis,RpDis18,xkDis,'pchip');
Rp24mapDis=interp1(x24Dis,RpDis24,xkDis,'pchip');
RpMapDis = [Rp6mapDis; Rp12mapDis; Rp18mapDis;
             Rp24mapDis];
RpMapDis = [RpMapDis(4,:); RpMapDis(3,:); RpMapDis(2,:);
            ; RpMapDis(1,:)]
figure(4)
plot (DxiMapDis, Rs24mapDis)

%Cp map
Cp6mapDis = interp1(x6Dis,CpDis6,xkDis,'pchip');
Cp12mapDis = interp1(x12Dis, CpDis12, xkDis, 'pchip');
Cp18mapDis = interp1(x18Dis, CpDis18, xkDis, 'pchip');
Cp24mapDis = interp1(x24Dis, CpDis24, xkDis, 'pchip');
CpMapDis = [Cp6mapDis; Cp12mapDis; Cp18mapDis;
            Cp24mapDis];
CpMapDis = [CpMapDis(4,:); CpMapDis(3,:); CpMapDis(2,:);
            ; CpMapDis(1,:)]

%DQb map
DxiMapDis = xkDis;

```

```

%I map
IMapDis = [-24; -18; -12; -6];
clearvars -except RsMapDis RpMapDis CpMapDis DxiMapDis
    IMapDis UocDis dUocdSoCDis
save modelValuesDis.mat
%% Charge
%Rs map
Rs6map=interp1(x6,Rs6,xk,'pchip');
Rs12map=interp1(x12,Rs12,xk,'pchip');
Rs18map=interp1(x18,Rs18,xk,'pchip');
Rs24map=interp1(x24,Rs24,xk,'pchip');
RsMap = [Rs6map; Rs12map; Rs18map; Rs24map];

plot (xk, Rs6map)
%Rp map
Rp6map=interp1(x6,Rp6,xk,'pchip');
Rp12map=interp1(x12,Rp12,xk,'pchip');
Rp18map=interp1(x18,Rp18,xk,'pchip');
Rp24map=interp1(x24,Rp24,xk,'pchip');
RpMap = [Rp6map; Rp12map; Rp18map; Rp24map];

%Cp map
Cp6map = interp1(x6,Cp6,xk,'pchip');
Cp12map = interp1(x12, Cp12, xk, 'pchip');
Cp18map = interp1(x18, Cp18, xk, 'pchip');
Cp24map = interp1(x24, Cp24, xk, 'pchip');
CpMap = [Cp6map; Cp12map; Cp18map; Cp24map];

RpMapDis = [RpMapDis(4,:); RpMapDis(3,:); RpMapDis(2,:)
    ; RpMapDis(1,:)]
RsMapDis = [RsMapDis(4,:); RsMapDis(3,:); RsMapDis(2,:)
    ; RsMapDis(1,:)]
CpMapDis = [CpMapDis(4,:); CpMapDis(3,:); CpMapDis(2,:)
    ; CpMapDis(1,:)]

Imap = [IMapDis; IMap];
Cpmap = [CpMapDis; CpMap];

```



```
Rsmmap = [RsMapDis; RsMap];
Rpmap = [RpMapDis; RpMap];

clearvars -except Rsmmap Rpmap Cpmap DxiMap Imap Uocmap
        dUocdSoC
save ltoMap.mat

%% fixing measured data
%Charge
ub6chg = [ub6(1:4830); ub6(4840:9666); ub6(9674:14500);
        ub6(14508:19335);ub6(19341:end)];
t6chg = [t6(1:4830); t12(4840:9666)-10; t6(9674:14500)
        -10-8; t6(14508:19335)-10-8-8;t6(19341:end)-10-8-8-6
        ];
ib6chg= [ib6(1:4830); ib6(4840:9666); ib6(9674:14500);
        ib6(14508:19335);ib6(19341:end)];

ub12chg = [ub12(1:3315); ub12(4000:7309); ub12(7520:end
        )];
t12chg = [t12(1:3315); t12(4000:7309)-685; t12(7520:end
        )-685-211];
ib12chg= [ib12(1:3315); ib12(4000:7309); ib12(7520:end)
        ];

ub18chg = [ub18(1:3317); ub18(3567:end)];
t18chg = [t18(1:3317); t18(3567:end)-250 ];
ib18chg= [ib18(1:3317); ib18(3567:end)];

%Discharge
ub6dis = [ub6Dis(1:6855); ub6Dis(6876:13725); ub6Dis
        (13765:end)];
t6dis = [t6Dis(1:6855); t6Dis(6876:13725)-21; t6Dis
        (13765:end)-21-40];
ib6dis = [ib6Dis(1:6855); ib6Dis(6876:13725); ib6Dis
        (13765:end)];
```

A.3 - Nelinearna regresija pomoću tangensa

```
function Uoc = tanReg(dxi,dUoc)

%Nelinearna regresija
% Uoc(kxi) = a*tan(b*kxi + c) + d
n = length(dxi);

%Izracun parametara S i T

S = zeros(1,n);
T = zeros(1,n);
for i=2:length(S)
    S(i) = S(i-1) + 1/2 * (dUoc(i)+dUoc(i-1))*(dxi(i)-
        dxi(i-1));
    T(i) = T(i-1) + 1/2 * ((dUoc(i))^2 + (dUoc(i-1))^2)
        *(dxi(i)-dxi(i-1));
end

% Izracun vrijednosti za matricu
T_sum = sum(T);
T2_sum = sum(T.*T);
T_S_sum = sum(T.*S);
T_x_sum = sum(T.*dxi);
T_y_sum = sum(T.*dUoc);

S_sum = sum(S);
S2_sum = sum(S.*S);
S_x_sum = sum(S.*dxi);
S_y_sum = sum(S.*dUoc);

x_sum = sum(dxi);
x2_sum = sum(dxi.*dxi);
x_y_sum = sum(dxi.*dUoc);

y_sum = sum(dUoc);

H = [T2_sum T_S_sum T_x_sum T_sum
     T_S_sum S2_sum S_x_sum S_sum
```

```

T_x_sum S_x_sum x2_sum x_sum
T_sum   S_sum   x_sum   n   ];

I = [ T_y_sum; S_y_sum; x_y_sum; y_sum ];

y1 = H\I;

A = y1(1);
B = y1(2);
C = y1(3);

%Odredivanje parametara b
b = 1/2*sqrt(-(B^2)+4*A*C);

X = zeros(1,n);
for i=1:length(X)
    X(i) = tan(b*dxi(i));
end

%Izracun vrijednosti za matricu
X_sum = sum(X);
X2_sum = sum(X.*X);
X_y_sum = sum(X.*dUoc);
X2_y_sum = sum(X.*X.*dUoc);
X2_y2_sum = sum(X.*X.*dUoc.*dUoc);

X_y2_sum = sum(X.*dUoc.*dUoc);

J = [ X2_y2_sum X2_y_sum X_y_sum
      X2_y_sum X2_sum X_sum
      X_y_sum X_sum n ];

K = [X_y2_sum; X_y_sum; y_sum];

y2 = J\K;

E = y2(1);
F = y2(2);
G = y2(3);

```

```
%Odredivanje parametara c
c = atan(E);

z11 = (tan(b*dxi+c)).^2;
z12 = (tan(b*dxi+c));
z13 = (tan(b*dxi+c)).*dUoc;

z11_sum = sum(z11);
z12_sum = sum(z12);
z13_sum = sum(z13);

L = [ z11_sum z12_sum
      z12_sum   n   ];
M = [z13_sum; y_sum];

y3 = L\M;

a = y3(1);
d = y3(2);

Uoc = a*tan(b*dxi + c) +d;
```

A.4 - Segmentna nelinearna regresija

```
function [xi, Uoc] = seg3(dxi, dUoc)

%Prvi set mjerenja
%dxi = [0 0.02579 0.05493 0.07692 0.099374 0.12821
        0.15385 0.23077 0.30769 0.38461 0.46154 0.53846
        0.61539 0.64103 0.66666 0.69231 0.71795 0.74359
        0.76923 0.79487 0.82051 0.84616 0.87180 0.89744
        0.92308 0.94872 0.97436 1];
%dUoc = [1.663190 2.068240 2.090715 2.100920 2.113350
         2.122406 2.131622 2.153261 2.152316 2.173648
         2.190853 2.212600 2.238120 2.249200 2.257410
         2.272650 2.287980 2.305020 2.323950 2.336490
         2.361500 2.385900 2.406090 2.438520 2.483400
         2.519110 2.642650 2.755360];

%Drugi set mjerenja

len_v = length(dxi);

x1 = dxi(1:5); y1 = dUoc(1:5);
x2_ = dxi(5:len_v-2); y2_ = dUoc(5:len_v-2);
x3 = dxi(len_v-2:len_v); y3 = dUoc(len_v-2:len_v);

dy1dx1 = (dUoc(6)-dUoc(5))/(dxi(6)-dxi(5));
dy3dx3 = (dUoc(len_v-3)-dUoc(len_v-2))/(dxi(len_v-3)-
        dxi(len_v-2));

x2 = [dxi(5) dxi(len_v-2)];
y2 = [dUoc(5) dUoc(len_v-2)];

xi_int = 0.0:0.001:1.0;
len_int = length(xi_int);

xi_int1 = xi_int(1:round(dxi(5)*len_int));
xi_int2 = xi_int(round(dxi(5)*len_int)+1:round(dxi(
        len_v-2)*len_int)-1);
xi_int3 = xi_int(round(dxi(len_v-2)*len_int):len_int);
```

```

A1 = [x1(1)^5 x1(1)^4 x1(1)^3 x1(1)^2 x1(1) 1;x1(2)^5
      x1(2)^4 x1(2)^3 x1(2)^2 x1(2) 1;x1(3)^5 x1(3)^4 x1
      (3)^3 x1(3)^2 x1(3) 1;x1(4)^5 x1(4)^4 x1(4)^3 x1(4)
      ^2 x1(4) 1;x1(5)^5 x1(5)^4 x1(5)^3 x1(5)^2 x1(5)
      1;5*x1(5)^4 4*x1(5)^3 3*x1(5)^2 2*x1(5) 1 0];
b1 = [y1(1);y1(2);y1(3);y1(4);y1(5);dy1dx1];

A1 = [x1(1)^5 x1(1)^4 x1(1)^3 x1(1)^2 x1(1) 1;x1(2)^5
      x1(2)^4 x1(2)^3 x1(2)^2 x1(2) 1;x1(3)^5 x1(3)^4 x1
      (3)^3 x1(3)^2 x1(3) 1;x1(4)^5 x1(4)^4 x1(4)^3 x1(4)
      ^2 x1(4) 1;x1(5)^5 x1(5)^4 x1(5)^3 x1(5)^2 x1(5)
      1;5*x1(5)^4 4*x1(5)^3 3*x1(5)^2 2*x1(5) 1 0];
b1 = [y1(1);y1(2);y1(3);y1(4);y1(5);dy1dx1];

A2 = [3*x2(1)^2 2*x2(1) 1 0; x2(1)^3 x2(1)^2 x2(1) 1;x2
      (2)^3 x2(2)^2 x2(2) 1;3*x2(2)^2 2*x2(2) 1 0];
b2 = [dy1dx1;y2(1);y2(2);dy3dx3];

A3 = [3*x3(1)^2 2*x3(1) 1 0; x3(1)^3 x3(1)^2 x3(1) 1;x3
      (2)^3 x3(2)^2 x3(2) 1;x3(3)^3 x3(3)^2 x3(3) 1];
b3 = [dy3dx3;y3(1);y3(2);y3(3)];

p1 = inv(A1)*b1;
p2 = inv(A2)*b2;
p3 = inv(A3)*b3;

y1_int = polyval(p1,xi_int1);
y2_int = polyval(p2,xi_int2);
y3_int = polyval(p3,xi_int3);

Uoc = [y1_int y2_int y3_int];
xi = [xi_int1 xi_int2 xi_int3];

end

figure(1),
plot(dxi,dUoc,'kd','LineWidth',1,'MarkerSize',6),hold

```

```
on
plot(xi_int1,y1_int,'r','LineWidth',2)
plot(xi_int2,y2_int,'g','LineWidth',2),
plot(xi_int3,y3_int,'b','LineWidth',2),grid on
legend('Mjerenje','Segment 1: polinom 5. reda','
      Segment 2: polinom 3. reda','Segment 3: polinom 3.
      reda')% xlabel('SoC'),ylabel('U_o_c [V]')

p1_der = (5:-1:1).*p1(1:5)';
p2_der = (3:-1:1).*p2(1:3)';
p3_der = (3:-1:1).*p3(1:3)';

dy1 = polyval(p1_der,xi_int1);
dy2 = polyval(p2_der,xi_int2);
dy3 = polyval(p3_der,xi_int3);

dy_vect = [dy1 dy2 dy3];

figure(2)
plot(xi_int,dy_vect,'b','LineWidth',2),grid on
xlabel('SoC'),ylabel('dU_o_c/dSoC [V]')
```

A.5 - Polinomna nelinearna regresija

```
function [fitresult, gof] = poly8(dxi, dUoc)
%CREATEFIT(DXI,DUOC)
% Create a fit.
%
% Data for 'Uoc_krivulja' fit:
%     X Input : dxi
%     Y Output: dUoc
% Output:
%     fitresult : a fit object representing the fit.
%     gof : structure with goodness-of fit info.
%
% See also FIT, CFIT, SFIT.

% Auto-generated by MATLAB on 23-Nov-2022 23:43:32

%% Fit: 'Uoc_krivulja'.
[xData, yData] = prepareCurveData( dxi, dUoc );

% Set up fittype and options.
ft = fittype( 'poly8' );
opts = fitoptions( 'Method', 'LinearLeastSquares' );
opts.Normalize = 'on';
opts.Robust = 'LAR';

% Fit model to data.
[fitresult, gof] = fit( xData, yData, ft, opts );

% Plot fit with data.
figure( 'Name', 'Uoc_krivulja' );
h = plot( fitresult, xData, yData );
legend( h, 'dUoc vs. dxi', 'Uoc_krivulja', 'Location',
        'NorthEast' );
% Label axes
xlabel dxi
ylabel dUoc
grid on
```


A.6 - Identifikacija parametara (6 A; Punjenje)

```
function [Rs, Rp, Tp, DQ] = dynMapChg6()

% Creating R0, Rp, tau 1D lookup table for 6A
%First set of measured data
%-----%
clc
%-----%

d = dir('*.*MAT'); %Finding all .MAT files
num_mat = length(d); % Checking the size of all MAT
    files
% calc_val = 18; %10 given points of input/output

%initialization of variables
% Rs = zeros(1,calc_val*num_mat+3);
% Rp = zeros(1,calc_val*num_mat+3);
% Tp = zeros(1,calc_val*num_mat+3);

Rs = [];
Rp = [];
Tp = [];
j = 1; %i value of the array

for i=1:num_mat

    filename = sprintf('DATA_%d.MAT',i); %saving as a
        variable

    load(filename) %loading in the chronical order of .
        MAT files

    % Sampling time

    Ts = t(2) - t(1);
```

```
% Detrending data
% Remove first 55 points to omit the initial
    voltage transient

if filename == 'DATA_1.MAT'

    %Nonlinear part%
    t_n = t(11:660); %Creating a initial value
    ub_n =ub(11:660);

    %Creating Uoc%
    [Uoc, p] = pwr2(t_n,ub_n);

    %Detrending the nonlinear part
    y = ub_n(1:530) - Uoc(1:530);
    u = detrend(ib(11:540),0);

    yyy = detrend(ub_n(531:end),1);
    uuu = detrend(ib(541:660),0);

    %Placing into one array
    ytot =[y; yyy];
    utot =[u; uuu];

    %Detrending linear part
    u = detrend(ib(11:660),0);

    yy1 = detrend(ub(661:2075),1);
    uu1 = detrend(ib(661:2075),0);

    yy2 = detrend(ub(2076:end-24),1);
    uu2 = detrend(ib(2076:end-24),0);

    %placing into one array

    yy = [yy1; yy2];
    uu = [uu1; uu2];

    %ARX model setup
```

```
na = 1; nb = 2; nk = 0;

Ts = t(2)-t(1);
%Nonlinear part

y1 = ytot(1:130); u1 = utot(1:130);
y2 = ytot(131:260); u2 = utot(131:260);
y3 = ytot(261:390); u3 = utot(261:390);
y4 = ytot(391:520); u4 = utot(391:520);
y5 = ytot(521:650); u5 = utot(521:650);
%Linear part
y6 = yy(1:830); u6 = uu(1:830);
y7 = yy(831:1660); u7 = uu(831:1660);
y8 = yy(1661:2490); u8 = uu(1661:2490);
y9 = yy(2491:3320); u9 = uu(2491:3320);
y10 = yy(3321:4150); u10 = uu(3321:4150);

% ARMAX model - ident. per data sets

% ARX model - ident. per data sets

thm1 = arx([y1 u1],[na nb nk]);
[num1,den1] = th2tf(thm1);
b1 = num1(1); b0 = num1(2); a = -den1(2);
Rs1 = b1; tau_p1 = -Ts/log(a); Rp1 = (b0 + a*b1
    )/(1-a);

thm2 = arx([y2 u2],[na nb nk]);
[num2,den2] = th2tf(thm2);
b1 = num2(1); b0 = num2(2); a = -den2(2);
Rs2 = b1; tau_p2 = -Ts/log(a); Rp2 = (b0 + a*b1
    )/(1-a);

thm3 = arx([y3 u3],[na nb nk]);
[num3,den3] = th2tf(thm3);
b1 = num3(1); b0 = num3(2); a = -den3(2);
Rs3 = b1; tau_p3 = -Ts/log(a); Rp3 = (b0 + a*b1
```

```
)/(1-a);

thm4 = arx([y4 u4],[na nb nk]);
[num4,den4] = th2tf(thm4);
b1 = num4(1); b0 = num4(2); a = -den4(2);
Rs4 = b1; tau_p4 = -Ts/log(a); Rp4 = (b0 + a*b1
)/(1-a);

thm5 = arx([y5 u5],[na nb nk]);
[num5,den5] = th2tf(thm5);
b1 = num5(1); b0 = num5(2); a = -den5(2);
Rs5 = b1; tau_p5 = -Ts/log(a); Rp5 = (b0 + a*b1
)/(1-a);

thm6 = arx([y6 u6],[na nb nk]);
[num6,den6] = th2tf(thm6);
b1 = num6(1); b0 = num6(2); a = -den6(2);
Rs6 = b1; tau_p6 = -Ts/log(a); Rp6 = (b0 + a*b1
)/(1-a);

thm7 = arx([y7 u7],[na nb nk]);
[num7,den7] = th2tf(thm7);
b1 = num7(1); b0 = num7(2); a = -den7(2);
Rs7 = b1; tau_p7 = -Ts/log(a); Rp7 = (b0 + a*b1
)/(1-a);

thm8 = arx([y8 u8],[na nb nk]);
[num8,den8] = th2tf(thm8);
b1 = num8(1); b0 = num8(2); a = -den8(2);
Rs8 = b1; tau_p8 = -Ts/log(a); Rp8 = (b0 + a*b1
)/(1-a);

thm9 = arx([y9 u9],[na nb nk]);
[num9,den9] = th2tf(thm9);
b1 = num9(1); b0 = num9(2); a = -den9(2);
Rs9 = b1; tau_p9 = -Ts/log(a); Rp9 = (b0 + a*b1
)/(1-a);

thm10 = arx([y10 u10],[na nb nk]);
```

```
[num10,den10] = th2tf(thm10);
b1 = num10(1); b0 = num10(2); a = -den10(2);
Rs10 = b1; tau_p10 = -Ts/log(a); Rp10 = (b0 + a
    *b1)/(1-a);

%temporary arrays

Rs_t = [Rs1 Rs2 Rs3 Rs4 Rs5 Rs6 Rs7 Rs8 Rs9
    Rs10];
Rp_t = [Rp1 Rp2 Rp3 Rp4 Rp5 Rp6 Rp7 Rp8 Rp9
    Rp10];
Tp_t = [tau_p1 tau_p2 tau_p3 tau_p4 tau_p5
    tau_p6 tau_p7 tau_p8 tau_p9 tau_p10];

l = length(Rs_t);

elseif filename == 'DATA_5.MAT'

%Nonlinear part%
t_n = t(11:end-20); %660
ub_n =ub(11:end-20);%660

tt = [];
Ib = [];
u0 = 2.4843;
SoC_ = SoC(11:end-20);
ib_ = ib(11:end-20);

for k = 1:length(SoC_)
    if ub_n(k) <= u0
        continue
    else
        Ub = ub_n(k:end);
        Ib = ib_(k:end);
        tt = 1:1:length(Ub);
        t_ = tt';
```

```
        break
    end
end
ub_ = Ub(301:end);
t__ = t_(301:end);

ub_lin = Ub(1:300);
t__lin = t_(1:300);
Uoc=four2(t__,ub_);

plot(t__,Uoc)
hold on
plot(t__,ub_)

%Detrending linear part
yyy = detrend(ub_lin,1);
uuu = detrend(Ib(1:300),0);

%Detrending Nonlinear part
yy = ub_ - Uoc;
uu = detrend(Ib(301:end),0);

y = [yyy; yy];
u = [uuu; uu];

plot(t_,y)

%ARX
na = 1; nb = 2; nk = 0;

%Sample time

Ts = t(2)-t(1);

y1 = y(1:190); u1 = u(1:190);
y2 = y(191:300); u2 = u(191:300);
y3 = y(301:570); u3 = u(301:570);
```

```
% ARX model - ident. per data sets

thm1 = arx([y1 u1],[na nb nk]);
[num1,den1] = th2tf(thm1);
b1 = num1(1); b0 = num1(2); a = -den1(2);
Rs1 = b1; tau_p1 = -Ts/log(a); Rp1 = (b0 + a*b1
    )/(1-a);

thm2 = arx([y2 u2],[na nb nk]);
[num2,den2] = th2tf(thm2);
b1 = num2(1); b0 = num2(2); a = -den2(2);
Rs2 = b1; tau_p2 = -Ts/log(a); Rp2 = (b0 + a*b1
    )/(1-a);

thm3 = arx([y3 u3],[na nb nk]);
[num3,den3] = th2tf(thm3);
b1 = num3(1); b0 = num3(2); a = -den3(2);
Rs3 = b1; tau_p3 = -Ts/log(a); Rp3 = (b0 + a*b1
    )/(1-a);

Rs_t = [Rs1 Rs2 Rs3];
Rp_t = [Rp1 Rp2 Rp3];
Tp_t = [tau_p1 tau_p2 tau_p3];

l = length(Rs_t);

else
    % Load data

    if i == 2
        u0 = 2.1684;
    elseif i == 3
        u0 = 2.2164;
    else
        u0 = 2.2287;
    end
end
```

```
% Sampling time
Ts = t(2) - t(1);
% Detrending data
% Remove first 55 points to omit the initial
    voltage transient

Ub_ = [];
tt = [];
Ib = [];

SoC_ = SoC(11:end);
ub_ = ub(11:end);
ib_ = ib(11:end);

for k = 1:length(SoC_)
    if ub_(k) <= u0
        continue
    else
        Ub = ub_(k:end);
        Ib = ib_(k:end);
        tt = 1:1:length(Ub);
        t_ = tt';
        break
    end
end

yy1 = detrend(Ub(1:480),1);
uu1 = detrend(Ib(1:480),0);

yy2 = detrend(Ub(481:960),1);
uu2 = detrend(Ib(481:960),0);

yy3 = detrend(Ub(961:1440),1);
uu3 = detrend(Ib(961:1440),0);
```



```
yy4 = detrend(Ub(1441:1920),1);
uu4 = detrend(Ib(1441:1920),0);

yy5 = detrend(Ub(1921:2400),1);
uu5 = detrend(Ib(1921:2400),0);

yy6 = detrend(Ub(2401:2880),1);
uu6 = detrend(Ib(2401:2880),0);

yy7 = detrend(Ub(2881:3360),1);
uu7 = detrend(Ib(2881:3360),0);

yy8 = detrend(Ub(3361:3840),1);
uu8 = detrend(Ib(3361:3840),0);

yy9 = detrend(Ub(3841:4320),1);
uu9 = detrend(Ib(3841:4320),0);

yy10 = detrend(Ub(4321:end-24),1);
uu10 = detrend(Ib(4321:end-24),0);

y = [yy1; yy2; yy3; yy4; yy5; yy6; yy7; yy8;
      yy9; yy10];
u = [uu1; uu2; uu3; uu4; uu5; uu6; uu7; uu8;
      uu9; uu10];

ylabel('y'), xlabel('t [s]')

na = 1; nb = 2; nk = 0;

% 10 data sets for identification

y1 = y(1:480); u1 = u(1:480);
y2 = y(481:960); u2 = u(481:960);
```

```
y3 = y(961:1440); u3 = u(961:1440);
y4 = y(1441:1920); u4 = u(1441:1920);
y5 = y(1921:2400); u5 = u(1921:2400);
y6 = y(2401:2880); u6 = u(2401:2880);
y7 = y(2881:3360); u7 = u(2881:3360);
y8 = y(3361:3840); u8 = u(3361:3840);
y9 = y(3841:4320); u9 = u(3841:4320);
y10 = y(4321:end); u10 = u(4321:end);

% ARX model - ident. per data sets

thm1 = arx([y1 u1],[na nb nk]);
[num1,den1] = th2tf(thm1);
b1 = num1(1); b0 = num1(2); a = -den1(2);
Rs1 = b1; tau_p1 = -Ts/log(a); Rp1 = (b0 + a*b1
    )/(1-a);

thm2 = arx([y2 u2],[na nb nk]);
[num2,den2] = th2tf(thm2);
b1 = num2(1); b0 = num2(2); a = -den2(2);
Rs2 = b1; tau_p2 = -Ts/log(a); Rp2 = (b0 + a*b1
    )/(1-a);

thm3 = arx([y3 u3],[na nb nk]);
[num3,den3] = th2tf(thm3);
b1 = num3(1); b0 = num3(2); a = -den3(2);
Rs3 = b1; tau_p3 = -Ts/log(a); Rp3 = (b0 + a*b1
    )/(1-a);

thm4 = arx([y4 u4],[na nb nk]);
[num4,den4] = th2tf(thm4);
b1 = num4(1); b0 = num4(2); a = -den4(2);
Rs4 = b1; tau_p4 = -Ts/log(a); Rp4 = (b0 + a*b1
    )/(1-a);

thm5 = arx([y5 u5],[na nb nk]);
[num5,den5] = th2tf(thm5);
```

```

b1 = num5(1); b0 = num5(2); a = -den5(2);
Rs5 = b1; tau_p5 = -Ts/log(a); Rp5 = (b0 + a*b1
    )/(1-a);

```

```

thm6 = arx([y6 u6],[na nb nk]);
[num6,den6] = th2tf(thm6);
b1 = num6(1); b0 = num6(2); a = -den6(2);
Rs6 = b1; tau_p6 = -Ts/log(a); Rp6 = (b0 + a*b1
    )/(1-a);

```

```

thm7 = arx([y7 u7],[na nb nk]);
[num7,den7] = th2tf(thm7);
b1 = num7(1); b0 = num7(2); a = -den7(2);
Rs7 = b1; tau_p7 = -Ts/log(a); Rp7 = (b0 + a*b1
    )/(1-a);

```

```

thm8 = arx([y8 u8],[na nb nk]);
[num8,den8] = th2tf(thm8);
b1 = num8(1); b0 = num8(2); a = -den8(2);
Rs8 = b1; tau_p8 = -Ts/log(a); Rp8 = (b0 + a*b1
    )/(1-a);

```

```

thm9 = arx([y9 u9],[na nb nk]);
[num9,den9] = th2tf(thm9);
b1 = num9(1); b0 = num9(2); a = -den9(2);
Rs9 = b1; tau_p9 = -Ts/log(a); Rp9 = (b0 + a*b1
    )/(1-a);

```

```

thm10 = arx([y10 u10],[na nb nk]);
[num10,den10] = th2tf(thm10);
b1 = num10(1); b0 = num10(2); a = -den10(2);
Rs10 = b1; tau_p10 = -Ts/log(a); Rp10 = (b0 + a
    *b1)/(1-a);

```

```

Rs_t = [Rs1 Rs2 Rs3 Rs4 Rs5 Rs6 Rs7 Rs8 Rs9
    Rs10 ];

```

```

Rp_t = [Rp1 Rp2 Rp3 Rp4 Rp5 Rp6 Rp7 Rp8 Rp9
    Rp10 ];

```

```

Tp_t = [tau_p1 tau_p2 tau_p3 tau_p4 tau_p5

```

```

        tau_p6 tau_p7 tau_p8 tau_p9 tau_p10];

    l = length(Rs_t);

end

%Inserting into true arrays
if i == 1;

    Rs(j:l) = Rs_t;
    Rp(j:l) = Rp_t;
    Tp(j:l) = Tp_t;
else

    Rs(j:j+1-1) = Rs_t;
    Rp(j:j+1-1) = Rp_t;
    Tp(j:j+1-1) = Tp_t;
end

j = j + 1;

clearvars -except Rs Rp Tp j i Ts filename
end
%Creating DQ map for following 1D lookup table

DQ1 = linspace(250,19000,4*10);
DQ2 = [250 500 570]+DQ1(end);
DQ = zeros(1,length(DQ1)+length(DQ2));
DQ_(1:40) = DQ1;
DQ_(41:43) = DQ2;
DQ = 6*DQ_/3600;

figure(3),
subplot(311),plot(DQ,1000*Rs,'b*:', 'LineWidth',1.25, '
    MarkerSize',6),grid on
ylabel('R_s [m\Omega]')
subplot(312),plot(DQ,1000*Rp,'b*:', 'LineWidth',1.25, '
    MarkerSize',6),grid on

```

```
ylabel('R_p [m\Omega]')
subplot(313),plot(DQ, Tp, 'b*:', 'LineWidth', 1.25, '
    MarkerSize', 6), grid on
ylabel('T_p [s]'), xlabel('\Delta Q [Ah]')

end
```

A.7 - Identifikacija parametara (6 A; Pražnjenje)

```
function [Rs, Rp, Tp, DQ] = dynMapDis6()

% Creating R0, Rp, tau 1D lookup table for 6A
%First set of measured data
%-----%
clc
%-----%

d = dir('*.*mat'); %Finding all .MAT files
num_mat = length(d); % Checking the size of all MAT
    files
% calc_val = 18; %10 given points of input/output

%initialization of variables
% Rs = zeros(1,calc_val*num_mat+3);
% Rp = zeros(1,calc_val*num_mat+3);
% Tp = zeros(1,calc_val*num_mat+3);

Rs = [];
Rp = [];
Tp = [];
j = 1; %i value of the array

for i=1:num_mat

    filename = sprintf('DATA_%d.mat',i); %saving as a
        variable

    load(filename) %loading in the chronological order of .
        MAT files

    % Sampling time

    Ts = t(2) - t(1);
```

```
% Detrending data
% Remove first 55 points to omit the initial
    voltage transient

if filename == 'DATA_1.mat'

    %Nonlinear part%
    t_n = t(2:401); %Creating a initial value
    ub_n =ub(2:401);

    %Linear part till ending
    ub_r = ub(402:6857);
    ib_r = ib(402:6857);

    %Creating Uoc%
    Uoc = pwr2(t_n,ub_n);

    figure(1)
    subplot(211),plot(t,ib),grid on
    ylabel('i_b [A]')
    subplot(212),plot(t,ub),grid on
    ylabel('u_b [V]'),xlabel('t [s]')

    % Sampling time
    Ts = t(2) - t(1);

    %Detrending the nonlinear part
    yy1 = ub_n - Uoc;
    uu1 = detrend(ib(2:401),0);

    yy2 = detrend(ub_r(1:660),1);
    uu2 = detrend(ib_r(1:660),0);

    yy3 = detrend(ub_r(661:1320),1);
    uu3 = detrend(ib_r(661:1320),0);
```

```
yy4 = detrend(ub_r(1321:1980),1);
uu4 = detrend(ib_r(1321:1980),0);

yy5 = detrend(ub_r(1981:2640),1);
uu5 = detrend(ib_r(1981:2640),0);

yy6 = detrend(ub_r(2641:3300),1);
uu6 = detrend(ib_r(2641:3300),0);

yy7 = detrend(ub_r(3301:3960),1);
uu7 = detrend(ib_r(3301:3960),0);

yy8 = detrend(ub_r(3961:4620),1);
uu8 = detrend(ib_r(3961:4620),0);

yy9 = detrend(ub_r(4621:5280),1);
uu9 = detrend(ib_r(4621:5280),0);

yy10 = detrend(ub_r(5281:5940),1);
uu10 = detrend(ib_r(5281:5940),0);

yy11 = detrend(ub_r(5941:end),1);
uu11 = detrend(ib_r(5941:end),0);

y = [yy1; yy2; yy3; yy4; yy5; yy6; yy7; yy8; yy9
      ; yy10; yy11];
u = [uu1; uu2; uu3; uu4; uu5; uu6; uu7; uu8; uu9
      ; uu10; uu11];
t_ = t(11:3315);

na = 1; nb = 2; nk = 0;

% Five data sets for identification
y1 = y(1:400); u1 = u(1:400);
y2 = y(401:1045); u2 = u(401:1045);
y3 = y(1046:1690); u3 = u(1046:1690);
y4 = y(1691:2335); u4 = u(1691:2335);
y5 = y(2336:2980); u5 = u(2336:2980);
```



```
y6 = y(2981:3625); u6 = u(2981:3625);
y7 = y(3626:4270); u7 = u(3626:4270);
y8 = y(4271:4915); u8 = u(4271:4915);
y9 = y(4916:5560); u9 = u(4916:5560);
y10 = y(5561:6205); u10 = u(5561:6205);
y11 = y(6206:end); u11 = u(6206:end);
% ARX model - ident. per data sets
thm1 = arx([y1 u1],[na nb nk]);
[num1,den1] = th2tf(thm1);
b1 = num1(1); b0 = num1(2); a = -den1(2);
Rs1 = b1; tau_p1 = -Ts/log(a); Rp1 = (b0 + a*b1
    )/(1-a);

thm2 = arx([y2 u2],[na nb nk]);
[num2,den2] = th2tf(thm2);
b1 = num2(1); b0 = num2(2); a = -den2(2);
Rs2 = b1; tau_p2 = -Ts/log(a); Rp2 = (b0 + a*b1
    )/(1-a);

thm3 = arx([y3 u3],[na nb nk]);
[num3,den3] = th2tf(thm3);
b1 = num3(1); b0 = num3(2); a = -den3(2);
Rs3 = b1; tau_p3 = -Ts/log(a); Rp3 = (b0 + a*b1
    )/(1-a);

thm4 = arx([y4 u4],[na nb nk]);
[num4,den4] = th2tf(thm4);
b1 = num4(1); b0 = num4(2); a = -den4(2);
Rs4 = b1; tau_p4 = -Ts/log(a); Rp4 = (b0 + a*b1
    )/(1-a);

thm5 = arx([y5 u5],[na nb nk]);
[num5,den5] = th2tf(thm5);
b1 = num5(1); b0 = num5(2); a = -den5(2);
Rs5 = b1; tau_p5 = -Ts/log(a); Rp5 = (b0 + a*b1
    )/(1-a);

thm6 = arx([y6 u6],[na nb nk]);
[num6,den6] = th2tf(thm6);
```

```

b1 = num6(1); b0 = num6(2); a = -den6(2);
Rs6 = b1; tau_p6 = -Ts/log(a); Rp6 = (b0 + a*b1
    )/(1-a);

```

```

thm7 = arx([y7 u7],[na nb nk]);
[num7,den7] = th2tf(thm7);
b1 = num7(1); b0 = num7(2); a = -den7(2);
Rs7 = b1; tau_p7 = -Ts/log(a); Rp7 = (b0 + a*b1
    )/(1-a);

```

```

thm8 = arx([y8 u8],[na nb nk]);
[num8,den8] = th2tf(thm8);
b1 = num8(1); b0 = num8(2); a = -den8(2);
Rs8 = b1; tau_p8 = -Ts/log(a); Rp8 = (b0 + a*b1
    )/(1-a);

```

```

thm9 = arx([y9 u9],[na nb nk]);
[num9,den9] = th2tf(thm9);
b1 = num9(1); b0 = num9(2); a = -den9(2);
Rs9 = b1; tau_p9 = -Ts/log(a); Rp9 = (b0 + a*b1
    )/(1-a);

```

```

thm10 = arx([y10 u10],[na nb nk]);
[num10,den10] = th2tf(thm10);
b1 = num10(1); b0 = num10(2); a = -den10(2);
Rs10 = b1; tau_p10 = -Ts/log(a); Rp10 = (b0 + a
    *b1)/(1-a);

```

```

thm11 = arx([y11 u11],[na nb nk]);
[num11,den11] = th2tf(thm11);
b1 = num11(1); b0 = num11(2); a = -den11(2);
Rs11 = b1; tau_p11 = -Ts/log(a); Rp11 = (b0 + a
    *b1)/(1-a);

```

```

%temporary arrays

```

```

Rs_t = [Rs1 Rs2 Rs3 Rs4 Rs5 Rs6 Rs7 Rs8 Rs9
    Rs10 Rs11];
Rp_t = [Rp1 Rp2 Rp3 Rp4 Rp5 Rp6 Rp7 Rp8 Rp9

```

```
        Rp10 Rp11];
Tp_t = [tau_p1 tau_p2 tau_p3 tau_p4 tau_p5
        tau_p6 tau_p7 tau_p8 tau_p9 tau_p10 tau_p11
        ];

l = length(Rs_t);

elseif filename == 'DATA_3.mat'

    Ts = t(2) - t(1);

    %Nonlinear part%
    t_n = t(4501:5792); %3280
    ub_n =ub(4501:5792); %3292
    ib_n = ib(4501:5792);
    ub_l = ub(1:4500);
    t_l = t(10:4500);
    tt = [];
    Ib = [];
    u0 = 2.132;
    SoC_ = DSoC(1:4500);
    ib_ = ib(1:4500);

    for k = 1:length(SoC_)
        if ub_l(k) >= u0
            continue
        else
            Ub = ub_l(k:end);
            Ib = ib_(k:end);
            tt = 1:1:length(Ub);
            t_ = tt';
            break
        end
    end

    Uoc= fourDis2(t_n,ub_n);
```

```
%Detrending linear part
yy1 = detrend(Ub(1:449),1);
uu1 = detrend(Ib(1:449),0);

yy2 = detrend(Ub(450:898),1);
uu2 = detrend(Ib(450:898),0);

yy3 = detrend(Ub(899:1347),1);
uu3 = detrend(Ib(899:1347),0);

yy4 = detrend(Ub(1348:1796),1);
uu4 = detrend(Ib(1348:1796),0);

yy5 = detrend(Ub(1797:2245),1);
uu5 = detrend(Ib(1797:2245),0);

yy6 = detrend(Ub(2246:2694),1);
uu6 = detrend(Ib(2246:2694),0);

yy7 = detrend(Ub(2695:3143),1);
uu7 = detrend(Ib(2695:3143),0);

yy8 = detrend(Ub(3144:3592),1);
uu8 = detrend(Ib(3144:3592),0);

yy9 = detrend(Ub(3593:4041),1);
uu9 = detrend(Ib(3593:4041),0);

yy10 = detrend(Ub(4042:end),1);
uu10 = detrend(Ib(4042:end),0);

yy11 = ub_n - Uoc;
uu11 = detrend(ib_n,0);
```

```
y = [yy1; yy2; yy3; yy4; yy5; yy6; yy7; yy8;  
     yy9; yy10; yy11];  
u = [uu1; uu2; uu3; uu4; uu5; uu6; uu7; uu8;  
     uu9; uu10; uu11];  
  
na = 1; nb = 2; nk = 0;  
  
% na = 1; nb = 2; nc = 1; nk = 0;  
  
% 10 data sets for identification  
  
y1 = y(1:449); u1 = u(1:449);  
y2 = y(450:898); u2 = u(450:898);  
y3 = y(899:1347); u3 = u(899:1347);  
y4 = y(1348:1796); u4 = u(1348:1796);  
y5 = y(1797:2245); u5 = u(1797:2245);  
y6 = y(2246:2694); u6 = u(2246:2694);  
y7 = y(2695:3143); u7 = u(2695:3143);  
y8 = y(3144:3592); u8 = u(3144:3592);  
y9 = y(3593:4041); u9 = u(3593:4041);  
y10 = y(4042:4494); u10 = u(4042:4494);  
y11 = y(4495:end); u11 = u(4495:end);  
  
thm1 = arx([y1 u1],[na nb nk]);  
[num1,den1] = th2tf(thm1);  
b1 = num1(1); b0 = num1(2); a = -den1(2);  
Rs1 = b1; tau_p1 = -Ts/log(a); Rp1 = (b0 + a*b1  
    )/(1-a);  
  
thm2 = arx([y2 u2],[na nb nk]);  
[num2,den2] = th2tf(thm2);  
b1 = num2(1); b0 = num2(2); a = -den2(2);  
Rs2 = b1; tau_p2 = -Ts/log(a); Rp2 = (b0 + a*b1  
    )/(1-a);  
  
thm3 = arx([y3 u3],[na nb nk]);
```

```
[num3,den3] = th2tf(thm3);
b1 = num3(1); b0 = num3(2); a = -den3(2);
Rs3 = b1; tau_p3 = -Ts/log(a); Rp3 = (b0 + a*b1
    )/(1-a);

thm4 = arx([y4 u4],[na nb nk]);
[num4,den4] = th2tf(thm4);
b1 = num4(1); b0 = num4(2); a = -den4(2);
Rs4 = b1; tau_p4 = -Ts/log(a); Rp4 = (b0 + a*b1
    )/(1-a);

thm5 = arx([y5 u5],[na nb nk]);
[num5,den5] = th2tf(thm5);
b1 = num5(1); b0 = num5(2); a = -den5(2);
Rs5 = b1; tau_p5 = -Ts/log(a); Rp5 = (b0 + a*b1
    )/(1-a);

thm6 = arx([y6 u6],[na nb nk]);
[num6,den6] = th2tf(thm6);
b1 = num6(1); b0 = num6(2); a = -den6(2);
Rs6 = b1; tau_p6 = -Ts/log(a); Rp6 = (b0 + a*b1
    )/(1-a);

thm7 = arx([y7 u7],[na nb nk]);
[num7,den7] = th2tf(thm7);
b1 = num7(1); b0 = num7(2); a = -den7(2);
Rs7 = b1; tau_p7 = -Ts/log(a); Rp7 = (b0 + a*b1
    )/(1-a);

thm8 = arx([y8 u8],[na nb nk]);
[num8,den8] = th2tf(thm8);
b1 = num8(1); b0 = num8(2); a = -den8(2);
Rs8 = b1; tau_p8 = -Ts/log(a); Rp8 = (b0 + a*b1
    )/(1-a);

thm9 = arx([y9 u9],[na nb nk]);
[num9,den9] = th2tf(thm9);
b1 = num9(1); b0 = num9(2); a = -den9(2);
Rs9 = b1; tau_p9 = -Ts/log(a); Rp9 = (b0 + a*b1
```

```

    ) / (1-a);

    thm10 = arx([y10 u10],[na nb nk]);
    [num10,den10] = th2tf(thm10);
    b1 = num10(1); b0 = num10(2); a = -den10(2);
    Rs10 = b1; tau_p10 = -Ts/log(a); Rp10 = (b0 + a
        *b1)/(1-a);

    thm11 = arx([y11 u11],[na nb nk]);
    [num11,den11] = th2tf(thm11);
    b1 = num11(1); b0 = num11(2); a = -den11(2);
    Rs11 = b1; tau_p11 = -Ts/log(a); Rp11 = (b0 + a
        *b1)/(1-a);

    Rs_t = [Rs1 Rs2 Rs3 Rs4 Rs5 Rs6 Rs7 Rs8 Rs9
        Rs10 Rs11 ];
    Rp_t = [Rp1 Rp2 Rp3 Rp4 Rp5 Rp6 Rp7 Rp8 Rp9
        Rp10 Rp11 ];
    Tp_t = [tau_p1 tau_p2 tau_p3 tau_p4 tau_p5
        tau_p6 tau_p7 tau_p8 tau_p9 tau_p10 tau_p11
        ];

    l = length(Rs_t);

else
    % Load data

    u0 = 2.2197;
    % Sampling time
    Ts = t(2) - t(1);
    % Detrending data
    % Remove first 55 points to omit the initial
        voltage transient

    tt = [];
    Ib = [];

```

```
SoC_ = DSoC(5:6855);
ub_ = ub(5:6855);
ib_ = ib(5:6855);

for k = 1:length(SoC_)
    if ub_(k) >= u0
        continue
    else
        Ub = ub_(k:end);
        Ib = ib_(k:end);
        tt = 1:1:length(Ub);
        t_ = tt';
        break
    end
end

yy1 = detrend(Ub(1:677),1);
uu1 = detrend(Ib(1:677),0);

yy2 = detrend(Ub(678:1354),1);
uu2 = detrend(Ib(678:1354),0);

yy3 = detrend(Ub(1355:2031),1);
uu3 = detrend(Ib(1355:2031),0);

yy4 = detrend(Ub(2032:2708),1);
uu4 = detrend(Ib(2032:2708),0);

yy5 = detrend(Ub(2709:3385),1);
uu5 = detrend(Ib(2709:3385),0);

yy6 = detrend(Ub(3386:4062),1);
uu6 = detrend(Ib(3386:4062),0);
```



```
yy7 = detrend(Ub(4063:4739),1);
uu7 = detrend(Ib(4063:4739),0);

yy8 = detrend(Ub(4740:5416),1);
uu8 = detrend(Ib(4740:5416),0);

yy9 = detrend(Ub(5417:6093),1);
uu9 = detrend(Ib(5417:6093),0);

yy10 = detrend(Ub(6094:end),1);
uu10 = detrend(Ib(6094:end),0);

y = [yy1; yy2; yy3; yy4; yy5; yy6; yy7; yy8;
      yy9; yy10];
u = [uu1; uu2; uu3; uu4; uu5; uu6; uu7; uu8;
      uu9; uu10];

ylabel('y'), xlabel('t [s]')

na = 1; nb = 2; nk = 0;

%          na = 1; nb = 2; nc = 1; nk = 0;

% 10 data sets for identification

y1 = y(1:677); u1 = u(1:677);
y2 = y(678:1354); u2 = u(678:1354);
y3 = y(1355:2031); u3 = u(1355:2031);
y4 = y(2032:2708); u4 = u(2032:2708);
y5 = y(2709:3385); u5 = u(2709:3385);
y6 = y(3386:4062); u6 = u(3386:4062);
y7 = y(4063:4739); u7 = u(4063:4739);
y8 = y(4740:5416); u8 = u(4740:5416);
y9 = y(5417:6093); u9 = u(5417:6093);
```

```
y10 = y(6094:end); u10 = u(6094:end);

thm1 = arx([y1 u1],[na nb nk]);
[num1,den1] = th2tf(thm1);
b1 = num1(1); b0 = num1(2); a = -den1(2);
Rs1 = b1; tau_p1 = -Ts/log(a); Rp1 = (b0 + a*b1
    )/(1-a);

thm2 = arx([y2 u2],[na nb nk]);
[num2,den2] = th2tf(thm2);
b1 = num2(1); b0 = num2(2); a = -den2(2);
Rs2 = b1; tau_p2 = -Ts/log(a); Rp2 = (b0 + a*b1
    )/(1-a);

thm3 = arx([y3 u3],[na nb nk]);
[num3,den3] = th2tf(thm3);
b1 = num3(1); b0 = num3(2); a = -den3(2);
Rs3 = b1; tau_p3 = -Ts/log(a); Rp3 = (b0 + a*b1
    )/(1-a);

thm4 = arx([y4 u4],[na nb nk]);
[num4,den4] = th2tf(thm4);
b1 = num4(1); b0 = num4(2); a = -den4(2);
Rs4 = b1; tau_p4 = -Ts/log(a); Rp4 = (b0 + a*b1
    )/(1-a);

thm5 = arx([y5 u5],[na nb nk]);
[num5,den5] = th2tf(thm5);
b1 = num5(1); b0 = num5(2); a = -den5(2);
Rs5 = b1; tau_p5 = -Ts/log(a); Rp5 = (b0 + a*b1
    )/(1-a);

thm6 = arx([y6 u6],[na nb nk]);
[num6,den6] = th2tf(thm6);
b1 = num6(1); b0 = num6(2); a = -den6(2);
Rs6 = b1; tau_p6 = -Ts/log(a); Rp6 = (b0 + a*b1
    )/(1-a);

thm7 = arx([y7 u7],[na nb nk]);
```

```

[num7,den7] = th2tf(thm7);
b1 = num7(1); b0 = num7(2); a = -den7(2);
Rs7 = b1; tau_p7 = -Ts/log(a); Rp7 = (b0 + a*b1
    )/(1-a);

thm8 = arx([y8 u8],[na nb nk]);
[num8,den8] = th2tf(thm8);
b1 = num8(1); b0 = num8(2); a = -den8(2);
Rs8 = b1; tau_p8 = -Ts/log(a); Rp8 = (b0 + a*b1
    )/(1-a);

thm9 = arx([y9 u9],[na nb nk]);
[num9,den9] = th2tf(thm9);
b1 = num9(1); b0 = num9(2); a = -den9(2);
Rs9 = b1; tau_p9 = -Ts/log(a); Rp9 = (b0 + a*b1
    )/(1-a);

thm10 = arx([y10 u10],[na nb nk]);
[num10,den10] = th2tf(thm10);
b1 = num10(1); b0 = num10(2); a = -den10(2);
Rs10 = b1; tau_p10 = -Ts/log(a); Rp10 = (b0 + a
    *b1)/(1-a);

Rs_t = [Rs1 Rs2 Rs3 Rs4 Rs5 Rs6 Rs7 Rs8 Rs9
    Rs10 ];
Rp_t = [Rp1 Rp2 Rp3 Rp4 Rp5 Rp6 Rp7 Rp8 Rp9
    Rp10 ];
Tp_t = [tau_p1 tau_p2 tau_p3 tau_p4 tau_p5
    tau_p6 tau_p7 tau_p8 tau_p9 tau_p10];

l = length(Rs_t);
end

%Inserting into true arrays
if i == 1;

Rs(j:l) = Rs_t;
Rp(j:l) = Rp_t;
Tp(j:l) = Tp_t;

```

```

else

    Rs(j:j+1-1) = Rs_t;
    Rp(j:j+1-1) = Rp_t;
    Tp(j:j+1-1) = Tp_t;
end

j = j + 1;

clearvars -except Rs Rp Tp j i Ts filename
end
%Creating DQ map for following 1D lookup table

DQ1 = 6*[400 1045 1690 2335 2980 3625 4270 4915 5560
        6205 6856]/3600;
DQ2 = 6*[677 1354 2031 2708 3385 4062 4739 5416 6093
        6778]/3600+ DQ1(end);
DQ3 = 6*[ 449 898 1347 1796 2245 2694 3143 3592 4041
        4494 5786]/3600 + DQ2(end);

DQ = [];
DQ = [DQ1 DQ2 DQ3];

figure(3),
subplot(311),plot(DQ,1000*Rs,'b*:', 'LineWidth',1.25, '
    MarkerSize',6),grid on
ylabel('R_s [m\Omega]')
subplot(312),plot(DQ,1000*Rp,'b*:', 'LineWidth',1.25, '
    MarkerSize',6),grid on
ylabel('R_p [m\Omega]')
subplot(313),plot(DQ,Tp,'b*:', 'LineWidth',1.25, '
    MarkerSize',6),grid on
ylabel('T_p [s]'),xlabel('\DeltaQ [Ah]')

end

```

A.8 - Identifikacija parametara (24 A; Punjenje)

```
function[Rs, Rp, Tp, DQ] = dynMapChg24()

% Creating R0, Rp, tau 1D lookup table for 6A
%First set of measured data
%-----%
clc
%-----%

% Load data
clear all
clc
%load DATA_1.MAT

load DATA003.MAT

%Nonlinear part - start%
t_n1 = t(3:150); %Creating a initial value
ub_n1 =ub(3:150);
ib_n1 = ib(3:150);
%Linear part till ending
ub_r = ub(151:4749);
ib_r = ib(151:4749);

%Nonlinear part - end%
ub_n2 = ub(4750:4922);
ib_n2 = ib(4750:4922);
t_n2 = t(4750:4922);

%Creating Uoc-start%
[Uoc1, p] = pwr2(t_n1,ub_n1);

%Creating Uoc-start%
Uoc2 = four2(t_n2, ub_n2);
```

```
% Sampling time
Ts = t(2) - t(1);

%Detrending the nonlinear part
yy1 = ub_n1 - Uoc1;
uu1 = detrend(ib_n1,0);

yy2 = detrend(ub_r(1:312),1);
uu2 = detrend(ib_r(1:312),0);

yy3 = detrend(ub_r(313:624),1);
uu3 = detrend(ib_r(313:624),0);

yy4 = detrend(ub_r(625:936),1);
uu4 = detrend(ib_r(625:936),0);

yy5 = detrend(ub_r(937:1248),1);
uu5 = detrend(ib_r(937:1248),0);

yy6 = detrend(ub_r(1249:1560),1);
uu6 = detrend(ib_r(1249:1560),0);

yy7 = detrend(ub_r(1561:1872),1);
uu7 = detrend(ib_r(1561:1872),0);

yy8 = detrend(ub_r(1873:2184),1);
uu8 = detrend(ib_r(1873:2184),0);

yy9 = detrend(ub_r(2185:2496),1);
uu9 = detrend(ib_r(2185:2496),0);

yy10 = detrend(ub_r(2497:2808),1);
uu10 = detrend(ib_r(2497:2808),0);

yy11 = detrend(ub_r(2809:3120),1);
uu11 = detrend(ib_r(2809:3120),0);

yy12 = detrend(ub_r(3121:3432),1);
uu12 = detrend(ib_r(3121:3432),0);
```

```
yy13 = detrend(ub_r(3433:3744),1);
uu13 = detrend(ib_r(3433:3744),0);

yy14 = detrend(ub_r(3745:4056),1);
uu14 = detrend(ib_r(3745:4056),0);

yy15 = detrend(ub_r(4057:4368),1);
uu15 = detrend(ib_r(4057:4368),0);

yy16 = detrend(ub_r(4369:end),1);
uu16 = detrend(ib_r(4369:end),0);

yy17 = ub_n2- Uoc2;
uu17 = detrend(ib_n2,0);

y=[yy1; yy2; yy3; yy4; yy5; yy6; yy7; yy8; yy9; yy10;
   yy11; yy12; yy13; yy14; yy15; yy16; yy17];
u=[uu1; uu2; uu3; uu4; uu5; uu6; uu7; uu8; uu9; uu10;
   uu11; uu12; uu13; uu14; uu15; uu16; uu17];
t_ = t(3:4922);

na = 1; nb = 2; nk = 0;

% Five data sets for identification
y1 = y(1:148); u1 = u(1:148);
y2 = y(149:608); u2 = u(149:608);
y3 = y(609:1068); u3 = u(609:1068);
y4 = y(1069:1528); u4 = u(1069:1528);
y5 = y(1529:1988); u5 = u(1529:1988);
y6= y(1989:2448); u6 = u(1989:2448);
y7 = y(2449:2908); u7 = u(2449:2908);
y8 = y(2909:3368); u8 = u(2909:3368);
y9 = y(3369:3828); u9 = u(3369:3828);
y10 = y(3829:4288); u10 = u(3829:4288);
y11 = y(4289:4747); u11 = u(4289:4747);
y12 = y(4748:end); u12 = u(4748:end);
```

```
thm1 = arx([y1 u1],[na nb nk]);  
[num1,den1] = th2tf(thm1);  
b1 = num1(1); b0 = num1(2); a = -den1(2);  
Rs1 = b1; tau_p1 = -Ts/log(a); Rp1 = (b0 + a*b1)/(1-a);
```

```
thm2 = arx([y2 u2],[na nb nk]);  
[num2,den2] = th2tf(thm2);  
b1 = num2(1); b0 = num2(2); a = -den2(2);  
Rs2 = b1; tau_p2 = -Ts/log(a); Rp2 = (b0 + a*b1)/(1-a);
```

```
thm3 = arx([y3 u3],[na nb nk]);  
[num3,den3] = th2tf(thm3);  
b1 = num3(1); b0 = num3(2); a = -den3(2);  
Rs3 = b1; tau_p3 = -Ts/log(a); Rp3 = (b0 + a*b1)/(1-a);
```

```
thm4 = arx([y4 u4],[na nb nk]);  
[num4,den4] = th2tf(thm4);  
b1 = num4(1); b0 = num4(2); a = -den4(2);  
Rs4 = b1; tau_p4 = -Ts/log(a); Rp4 = (b0 + a*b1)/(1-a);
```

```
thm5 = arx([y5 u5],[na nb nk]);  
[num5,den5] = th2tf(thm5);  
b1 = num5(1); b0 = num5(2); a = -den5(2);  
Rs5 = b1; tau_p5 = -Ts/log(a); Rp5 = (b0 + a*b1)/(1-a);
```

```
thm6 = arx([y6 u6],[na nb nk]);  
[num6,den6] = th2tf(thm6);  
b1 = num6(1); b0 = num6(2); a = -den6(2);  
Rs6 = b1; tau_p6 = -Ts/log(a); Rp6 = (b0 + a*b1)/(1-a);
```

```
thm7 = arx([y7 u7],[na nb nk]);  
[num7,den7] = th2tf(thm7);  
b1 = num7(1); b0 = num7(2); a = -den7(2);  
Rs7 = b1; tau_p7 = -Ts/log(a); Rp7 = (b0 + a*b1)/(1-a);
```

```
thm8 = arx([y8 u8],[na nb nk]);  
[num8,den8] = th2tf(thm8);  
b1 = num8(1); b0 = num8(2); a = -den8(2);
```



```

Rs8 = b1; tau_p8 = -Ts/log(a); Rp8 = (b0 + a*b1)/(1-a);

thm9 = arx([y9 u9],[na nb nk]);
[num9,den9] = th2tf(thm9);
b1 = num9(1); b0 = num9(2); a = -den9(2);
Rs9 = b1; tau_p9 = -Ts/log(a); Rp9 = (b0 + a*b1)/(1-a);

thm10 = arx([y10 u10],[na nb nk]);
[num10,den10] = th2tf(thm10);
b1 = num10(1); b0 = num10(2); a = -den10(2);
Rs10 = b1; tau_p10 = -Ts/log(a); Rp10 = (b0 + a*b1)/(1-
    a);

thm11 = arx([y11 u11],[na nb nk]);
[num11,den11] = th2tf(thm11);
b1 = num11(1); b0 = num11(2); a = -den11(2);
Rs11 = b1; tau_p11 = -Ts/log(a); Rp11 = (b0 + a*b1)/(1-
    a);

thm12 = arx([y12 u12],[na nb nk]);
[num12,den12] = th2tf(thm12);
b1 = num12(1); b0 = num12(2); a = -den12(2);
Rs12 = b1; tau_p12 = -Ts/log(a); Rp12 = (b0 + a*b1)/(1-
    a);

Rs = [Rs1 Rs2 Rs3 Rs4 Rs5 Rs6 Rs7 Rs8 Rs9 Rs10 Rs11
    Rs12 ];
Rp = [Rp1 Rp2 Rp3 Rp4 Rp5 Rp6 Rp7 Rp8 Rp9 Rp10 Rp11
    Rp12 ];
Tp = [tau_p1 tau_p2 tau_p3 tau_p4 tau_p5 tau_p6 tau_p7
    tau_p8 tau_p9 tau_p10 tau_p11 tau_p12];

%Creating DQ map for following 1D lookup table

DQ = 24*[148 608 1068 1528 1988 2448 2908 3368 3828
    4288 4747 4920]/3600;

figure(3),
subplot(311),plot(DQ,1000*Rs,'b*:', 'LineWidth',1.5, '

```

```
    MarkerSize',6),grid on
ylabel('R_s [m\Omega]')
subplot(312),plot(DQ,1000*Rp,'b*:', 'LineWidth',1.5, '
    MarkerSize',6),grid on
ylabel('R_p [m\Omega]')
subplot(313),plot(DQ, Tp, 'b*:', 'LineWidth',1.5, '
    MarkerSize',6),grid on
ylabel('T_p [s]'),xlabel('\DeltaQ [Ah]')
```

end

A.9 - Identifikacija parametara (24 A; Pražnjenje)

```
function[Rs, Rp, Tp, DQ] = dynMapDis24()

% Creating R0, Rp, tau 1D lookup table for 6A
%First set of measured data
%-----%
clc
%-----%

% Load data
clear all
clc

load DATA003.MAT

%Nonlinear part - start%
t_n1 = t(3:150); %Creating a initial value
ub_n1 =ub(3:150);
ib_n1 = ib(3:150);
%Linear part till ending
ub_r = ub(151:4000);
ib_r = ib(151:4000);
t_r = t(151:4000);
plot(t_r, ub_r)

%Nonlinear part - end%
ub_n2 = ub(4001:4648);
ib_n2 = ib(4001:4648);
t_n2 = t(4001:4648);

%Creating Uoc-start%
Uoc1 = pwr2(t_n1,ub_n1);

%Creating Uoc-end%
Uoc2 = fourDis2(t_n2, ub_n2);
```

```
plot(t_n2, ub_n2)
hold on
plot(t_n2, Uoc2)

% Sampling time
Ts = t(2) - t(1);

%Detrending the nonlinear part
yy1 = ub_n1 - Uoc1;
uu1 = detrend(ib_n1,0);

yy2 = detrend(ub_r(1:275),1);
uu2 = detrend(ib_r(1:275),0);

yy3 = detrend(ub_r(276:550),1);
uu3 = detrend(ib_r(276:550),0);

yy4 = detrend(ub_r(551:825),1);
uu4 = detrend(ib_r(551:825),0);

yy5 = detrend(ub_r(826:1100),1);
uu5 = detrend(ib_r(826:1100),0);

yy6 = detrend(ub_r(1101:1375),1);
uu6 = detrend(ib_r(1101:1375),0);

yy7 = detrend(ub_r(1376:1650),1);
uu7 = detrend(ib_r(1376:1650),0);

yy8 = detrend(ub_r(1651:1925),1);
uu8 = detrend(ib_r(1651:1925),0);

yy9 = detrend(ub_r(1926:2220),1);
uu9 = detrend(ib_r(1926:2220),0);

yy10 = detrend(ub_r(2221:2475),1);
uu10 = detrend(ib_r(2221:2475),0);
```

```
yy11 = detrend(ub_r(2476:2750),1);
uu11 = detrend(ib_r(2476:2750),0);

yy12 = detrend(ub_r(2751:3025),1);
uu12 = detrend(ib_r(2751:3025),0);

yy13 = detrend(ub_r(3026:3300),1);
uu13 = detrend(ib_r(3026:3300),0);

yy14 = detrend(ub_r(3301:3575),1);
uu14 = detrend(ib_r(3301:3575),0);

yy15 = detrend(ub_r(3576:end),1);
uu15 = detrend(ib_r(3576:end),0);

yy16 = ub_n2- Uoc2;
uu16 = detrend(ib_n2,0);

y =[yy1; yy2; yy3; yy4; yy5; yy6; yy7; yy8; yy9; yy10;
    yy11; yy12; yy13; yy14; yy15; yy16];
u =[uu1; uu2; uu3; uu4; uu5; uu6; uu7; uu8; uu9; uu10;
    uu11; uu12; uu13; uu14; uu15; uu16];

t_ = t(3:4648);

na = 1; nb = 2; nk = 0;

% Five data sets for identification
y1 = y(1:148); u1 = u(1:148);
y2 = y(149:608); u2 = u(149:608);
y3 = y(609:1068); u3 = u(609:1068);
y4 = y(1069:1528); u4 = u(1069:1528);
y5 = y(1529:1988); u5 = u(1529:1988);
y6= y(1989:2448); u6 = u(1989:2448);
y7 = y(2449:2908); u7 = u(2449:2908);
y8 = y(2909:3368); u8 = u(2909:3368);
y9 = y(3369:3828); u9 = u(3369:3828);
```

```
y10 = y(3829:4000); u10 = u(3829:4000);
y11 = y(4001:4323); u11 = u(4001:4323);
y12 = y(4324:end); u12 = u(4324:end);

thm1 = arx([y1 u1],[na nb nk]);
[num1,den1] = th2tf(thm1);
b1 = num1(1); b0 = num1(2); a = -den1(2);
Rs1 = b1; tau_p1 = -Ts/log(a); Rp1 = (b0 + a*b1)/(1-a);

thm2 = arx([y2 u2],[na nb nk]);
[num2,den2] = th2tf(thm2);
b1 = num2(1); b0 = num2(2); a = -den2(2);
Rs2 = b1; tau_p2 = -Ts/log(a); Rp2 = (b0 + a*b1)/(1-a);

thm3 = arx([y3 u3],[na nb nk]);
[num3,den3] = th2tf(thm3);
b1 = num3(1); b0 = num3(2); a = -den3(2);
Rs3 = b1; tau_p3 = -Ts/log(a); Rp3 = (b0 + a*b1)/(1-a);

thm4 = arx([y4 u4],[na nb nk]);
[num4,den4] = th2tf(thm4);
b1 = num4(1); b0 = num4(2); a = -den4(2);
Rs4 = b1; tau_p4 = -Ts/log(a); Rp4 = (b0 + a*b1)/(1-a);

thm5 = arx([y5 u5],[na nb nk]);
[num5,den5] = th2tf(thm5);
b1 = num5(1); b0 = num5(2); a = -den5(2);
Rs5 = b1; tau_p5 = -Ts/log(a); Rp5 = (b0 + a*b1)/(1-a);

thm6 = arx([y6 u6],[na nb nk]);
[num6,den6] = th2tf(thm6);
b1 = num6(1); b0 = num6(2); a = -den6(2);
Rs6 = b1; tau_p6 = -Ts/log(a); Rp6 = (b0 + a*b1)/(1-a);

thm7 = arx([y7 u7],[na nb nk]);
[num7,den7] = th2tf(thm7);
b1 = num7(1); b0 = num7(2); a = -den7(2);
Rs7 = b1; tau_p7 = -Ts/log(a); Rp7 = (b0 + a*b1)/(1-a);
```

```

thm8 = arx([y8 u8],[na nb nk]);
[num8,den8] = th2tf(thm8);
b1 = num8(1); b0 = num8(2); a = -den8(2);
Rs8 = b1; tau_p8 = -Ts/log(a); Rp8 = (b0 + a*b1)/(1-a);

thm9 = arx([y9 u9],[na nb nk]);
[num9,den9] = th2tf(thm9);
b1 = num9(1); b0 = num9(2); a = -den9(2);
Rs9 = b1; tau_p9 = -Ts/log(a); Rp9 = (b0 + a*b1)/(1-a);

thm10 = arx([y10 u10],[na nb nk]);
[num10,den10] = th2tf(thm10);
b1 = num10(1); b0 = num10(2); a = -den10(2);
Rs10 = b1; tau_p10 = -Ts/log(a); Rp10 = (b0 + a*b1)/(1-
    a);

thm11 = arx([y11 u11],[na nb nk]);
[num11,den11] = th2tf(thm11);
b1 = num11(1); b0 = num11(2); a = -den11(2);
Rs11 = b1; tau_p11 = -Ts/log(a); Rp11 = (b0 + a*b1)/(1-
    a);

thm12 = arx([y12 u12],[na nb nk]);
[num12,den12] = th2tf(thm12);
b1 = num12(1); b0 = num12(2); a = -den12(2);
Rs12 = b1; tau_p12 = -Ts/log(a); Rp12 = (b0 + a*b1)/(1-
    a);

Rs = [Rs1 Rs2 Rs3 Rs4 Rs5 Rs6 Rs7 Rs8 Rs9 Rs10 Rs11
    Rs12 ];
Rp = [Rp1 Rp2 Rp3 Rp4 Rp5 Rp6 Rp7 Rp8 Rp9 Rp10 Rp11
    Rp12 ];
Tp = [tau_p1 tau_p2 tau_p3 tau_p4 tau_p5 tau_p6 tau_p7
    tau_p8 tau_p9 tau_p10 tau_p11 tau_p12];

%Creating DQ map for following 1D lookup table

DQ = 24*[148 608 1068 1528 1988 2448 2908 3368 3828

```

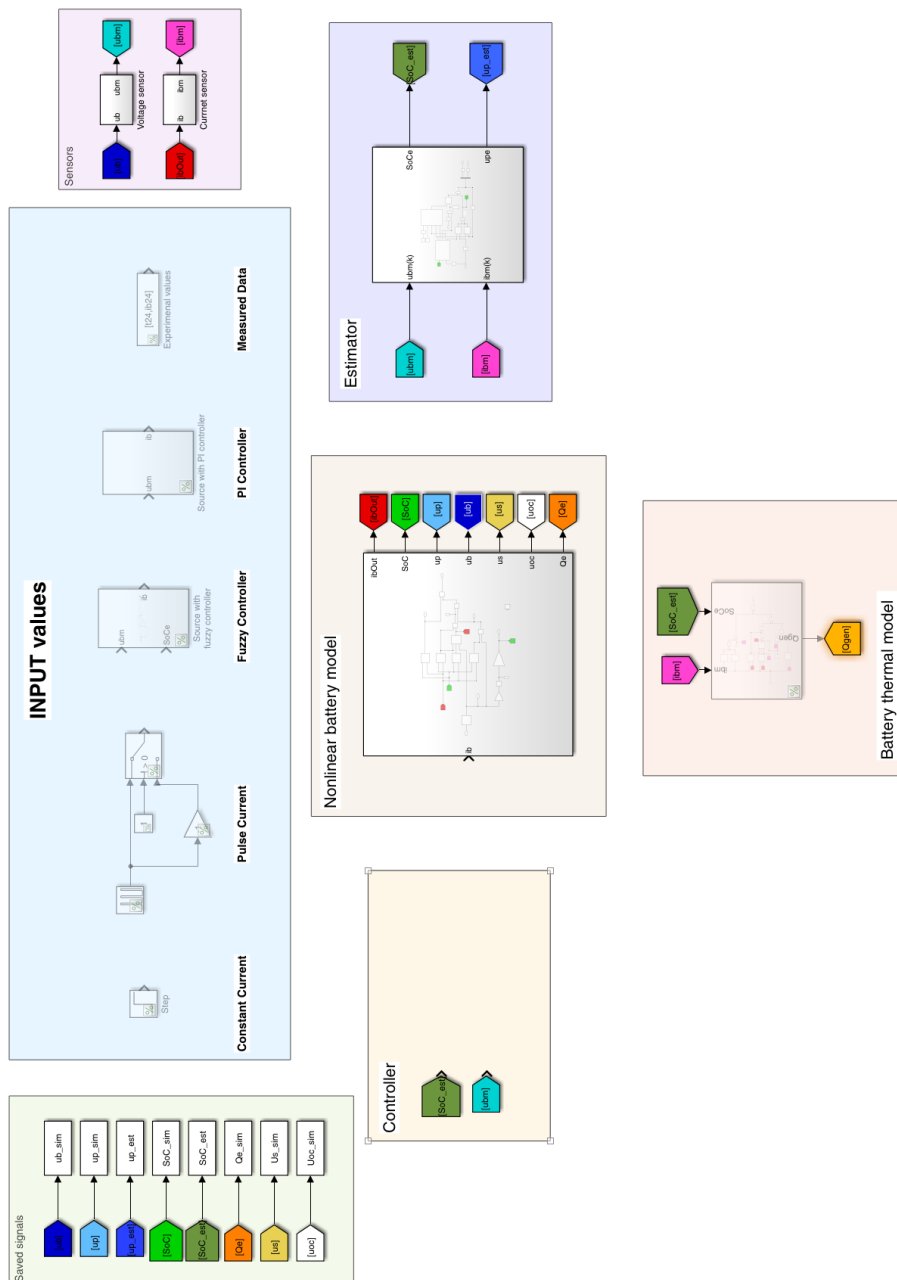
```
4000 4323 4646]/3600;

figure(3),
subplot(311),plot(DQ,1000*Rs,'b*:', 'LineWidth',1.5, '
    MarkerSize',6),grid on
ylabel('R_s [m\Omega]')
subplot(312),plot(DQ,1000*Rp,'b*:', 'LineWidth',1.5, '
    MarkerSize',6),grid on
ylabel('R_p [m\Omega]')
subplot(313),plot(DQ, Tp, 'b*:', 'LineWidth',1.5, '
    MarkerSize',6),grid on
ylabel('T_p [s]'),xlabel('\Delta Q [Ah]')

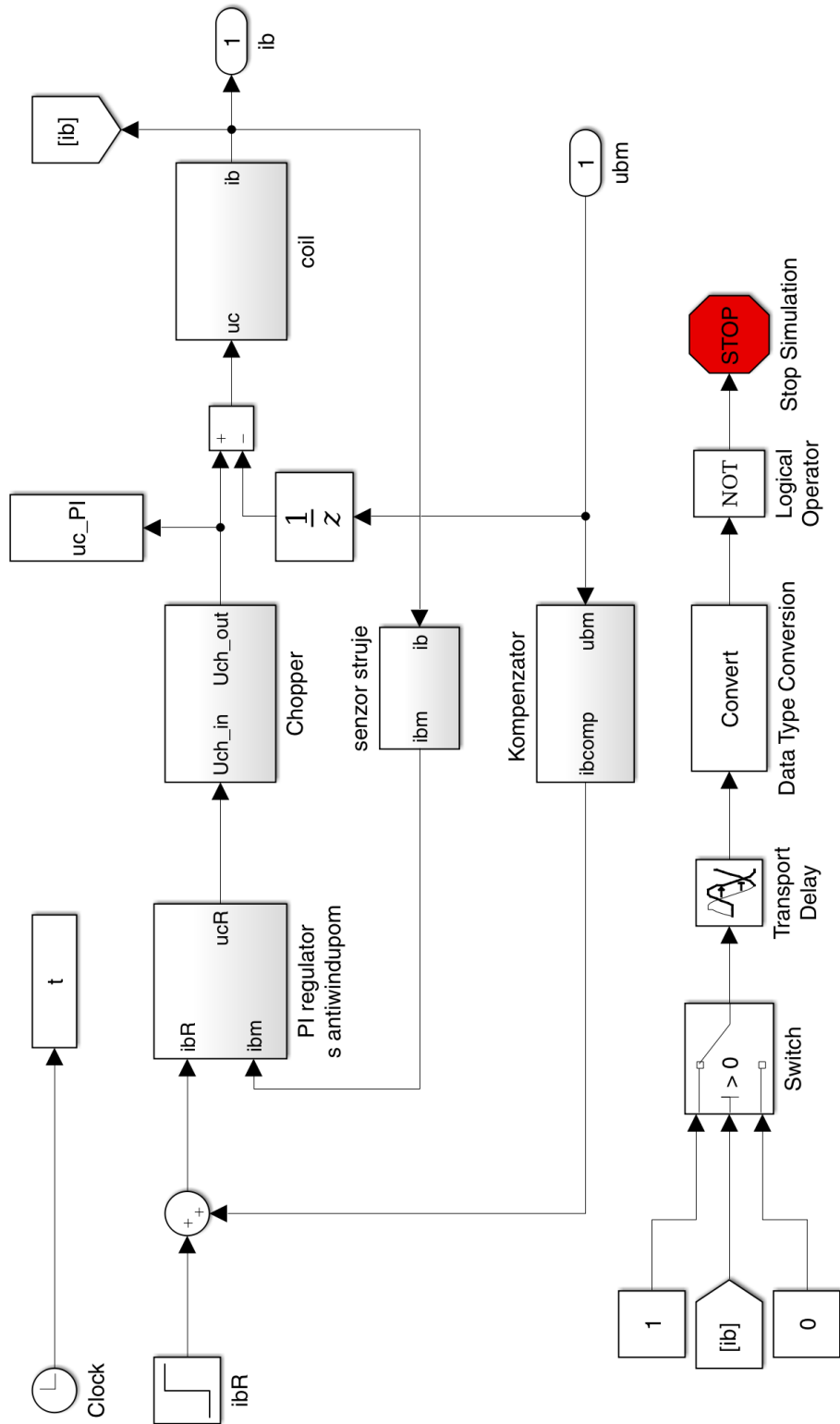
end
```


Dodatak B

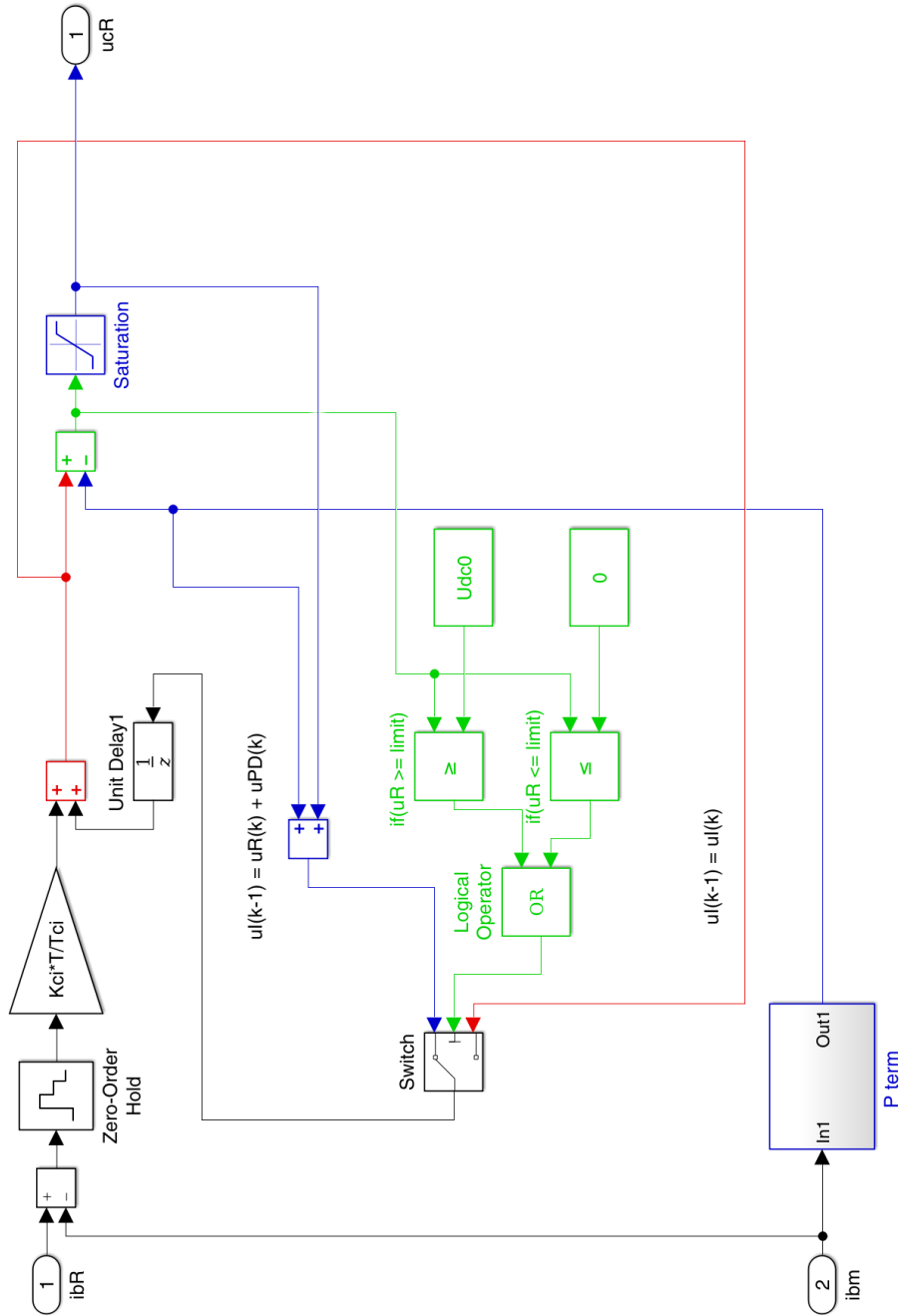
B.1 - Ukupni model



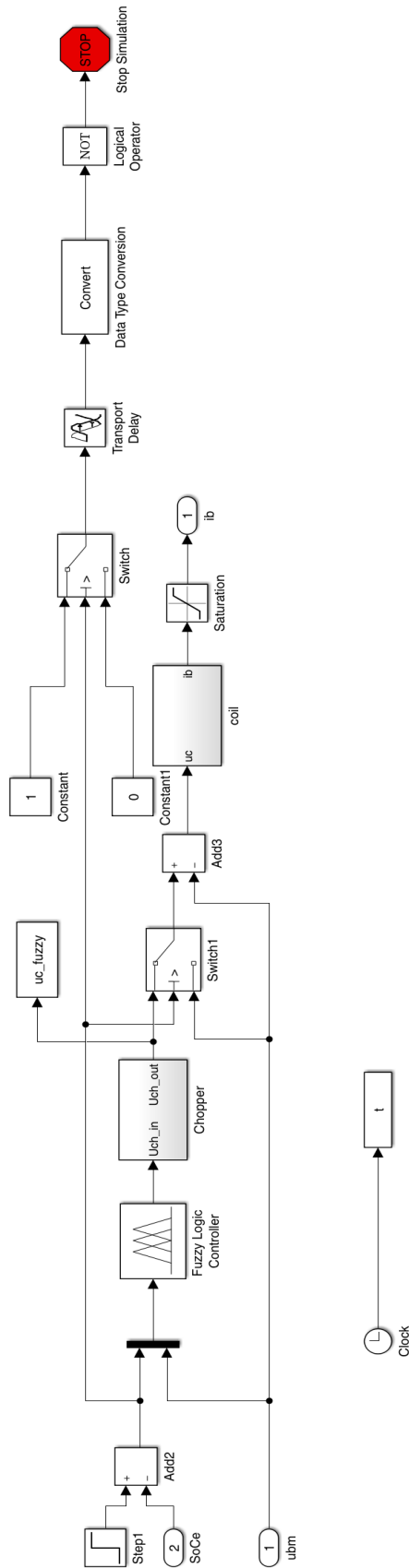
B.2 - PI regulator sa podređenim regulacijskim krugom



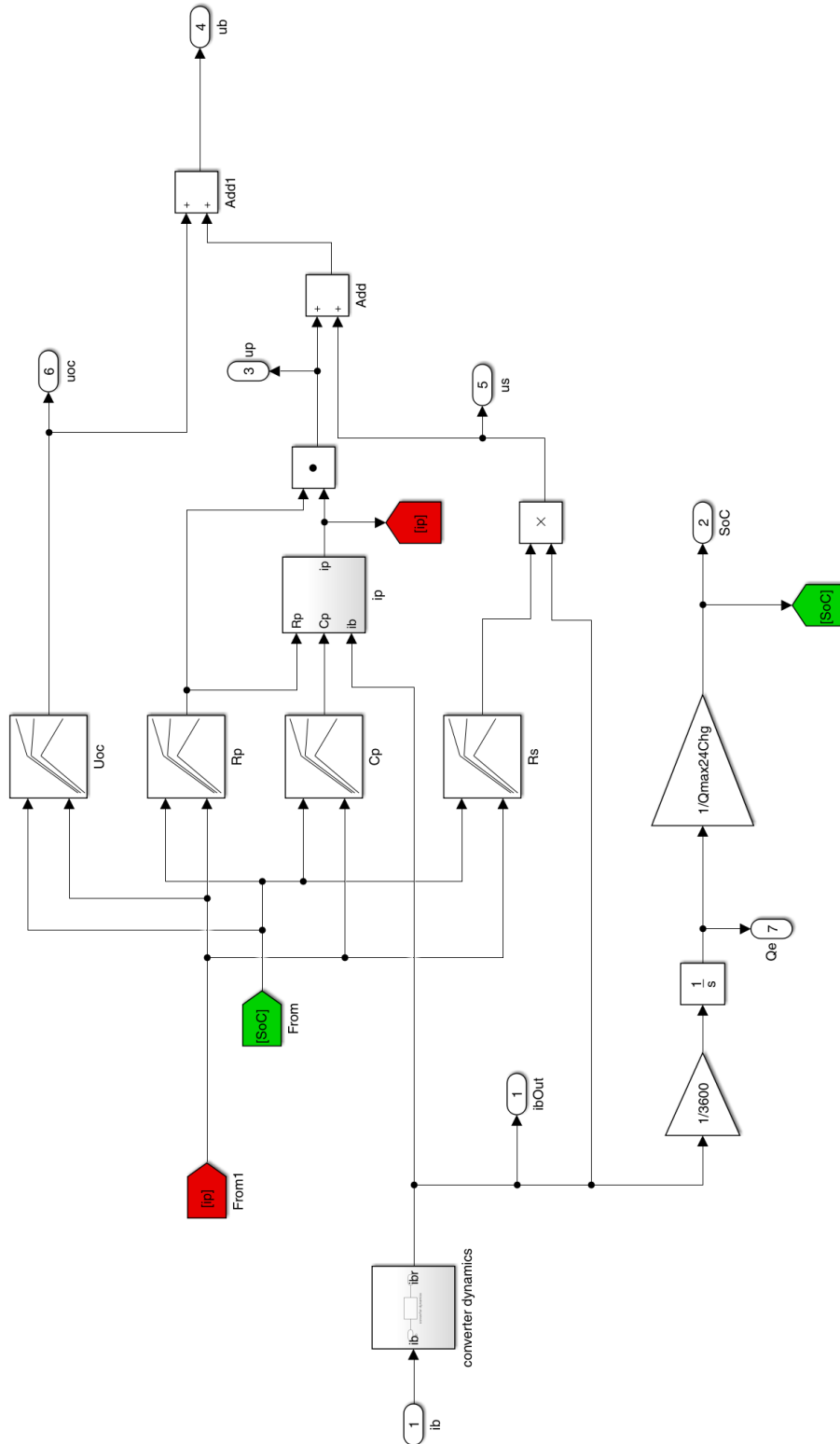
B.3 - PI regulator



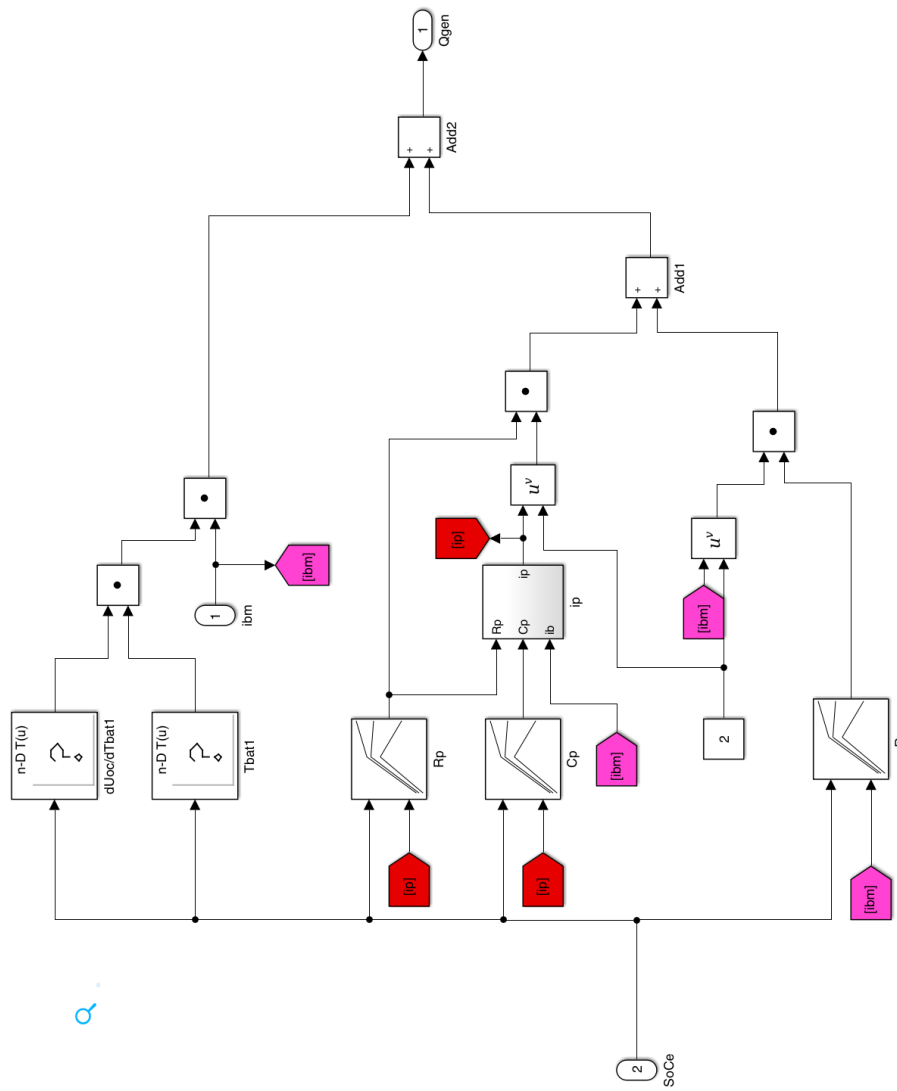
B.4 - Neizraziti regulator



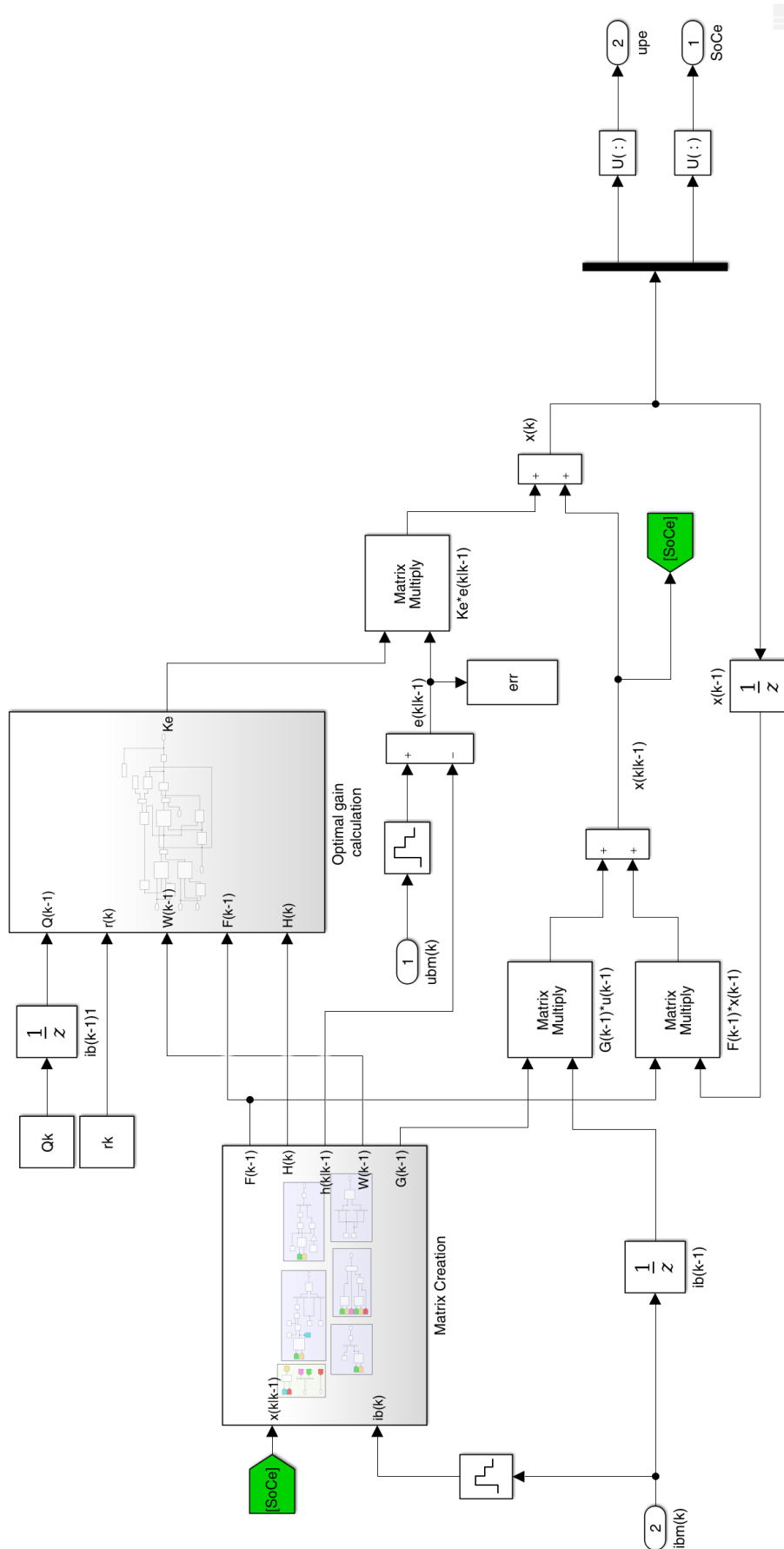
B.5 - Model baterije



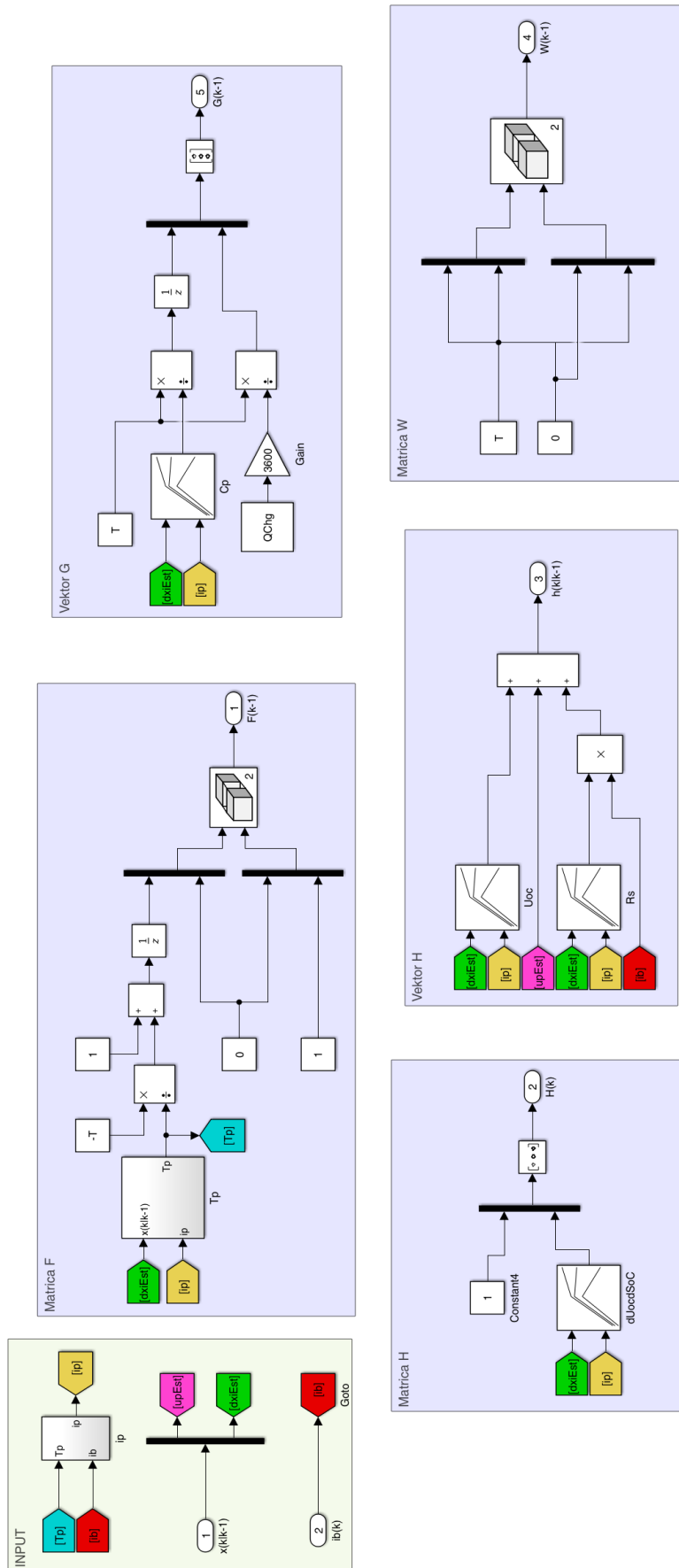
B.6 - Termalni model



B.7 - Estimator stanja



B.8 - Model izračun matrica prostora stanja



B.9 - Ricattijev model

