

Univerza v Ljubljani
Fakulteta za računalništvo in informatiko

Matjaž Okretič

**Mehki algoritem segrevanja in vzdrževanja
temperature v procesu varjenja piva**

DIPLOMSKA NALOGA NA UNIVERZITETNEM ŠTUDIJU

doc. dr. Iztok Lebar Bajec
MENTOR

Ljubljana, 2013

© 2013, Univerza v Ljubljani, Fakulteta za računalništvo in informatiko

Rezultati diplomskega dela so intelektualna lastnina Fakultete za računalništvo in informatiko Univerze v Ljubljani. Za objavlanje ali izkoriščanje rezultatov diplomskega dela je potrebno pisno soglasje Fakultete za računalništvo in informatiko ter mentorja.



Št. naloge: 01882/2012

Datum: 03.12.2012

Univerza v Ljubljani, Fakulteta za računalništvo in informatiko izdaja naslednjo nalogo:

Kandidat: **MATJAŽ OKRETIČ**

Naslov: **MEHKI ALGORITEM SEGREVANJA IN VZDRŽEVANJA
TEMPERATURE V PROCESU VARJENJA PIVA
FUZZY LOGIC BASED HEATING AND TEMPERATURE
MAINTENANCE ALGORITHM FOR THE BEER BREWING PROCESS**

Vrsta naloge: Diplomsko delo univerzitetnega študija

Tematika naloge:

Pivo je tudi v lokalnem okolju splošno znana in priljubljena alkoholna pijača, zvarjena iz poljščin, ki vsebujejo škrob, in občasno obogatena z dodatkom različnih začimb. Na Ljubljanskem barju so našli 3900 let stare pivske posode. Vsekakor ima varjenje piva na območju Slovenije že bogato zgodovino. Kot računalničarji v recepture varjenja piva (še) ne moremo posegati, vendar lahko varilcem pomagamo z avtomatizacijo in krmiljenjem nekaterih segmentov v procesu varjenja piva.

V diplomski nalogi proučite obstoječi algoritem segrevanja in vzdrževanja temperature v procesu varjenja piva. Osredotočite se predvsem na porabo energije, drsenje temperature v fazi vzdrževanja ter doseganje zelenih temperaturnih odstopanj. Z uporabo mehke logike nato realizirajte algoritem, ki bo odpravil pomanjkljivosti obstoječega algoritma.

Mentor:

doc. dr. Iztok Lebar Bajec



Dekan:

prof. dr. Nikolaj Zimic

IZJAVA O AVTORSTVU DIPLOMSKEGA DELA

Spodaj podpisani izjavljam, da sem avtor dela, da slednje ne vsebuje materiala, ki bi ga kdorkoli predhodno že objavil ali oddal v obravnavo za pridobitev naziva na univerzi ali drugem visokošolskem zavodu, razen v primerih kjer so navedeni viri.

S svojim podpisom zagotavljam, da:

- sem delo izdelal samostojno pod mentorstvom doc. dr. Iztoka Lebarja Bajca,
- so elektronska oblika dela, naslov (slov., angl.), povzetek (slov., angl.) ter ključne besede (slov., angl.) identični s tiskano obliko in
- soglašam z javno objavo elektronske oblike dela v zbirki "Dela FRI".

— Matjaž Okretič, Ljubljana, januar 2013.

Univerza v Ljubljani
Fakulteta za računalništvo in informatiko

Matjaž Okretič

Mehki algoritem segrevanja in vzdrževanja temperature v procesu varjenja piva

POVZETEK

Diplomsko delo predstavlja izboljšavo algoritma, ki upravlja število delujočih grelcev pri segrevanju in vzdrževanju temperature v procesu varjenja piva, z uporabo mehke logike. Cilji novega algoritma so manjša časovna in energijska poraba ter manjše odstopanje končne temperature od želene temperature. Koraki potrebni za razvoj izboljšane algoritma so razdeljeni v več poglavij.

Najprej je predstavljeno okolje na katerem algoritem deluje. Opisan je kotel in njegovi elementi, ki so pomembni za segrevanje in vzdrževanje temperature. Prav tako je opisan postopek celotnega procesa kuhanja piva in obstoječi algoritem, katerega rezultate je potrebno izboljšati. V tem poglavju so predstavljena tudi orodja uporabljena pri razvoju.

Drugo poglavje je namenjeno ugotavljanju lastnosti okolja. Gre za zajemanje podatkov okolja, ki se nanašajo na temperature in število delujočih grelcev v različnih možnih scenarijih ter njihova interpretacija. Podatki pridobljeni v tem poglavju so osnova za nadaljnji razvoj algoritma.

Opis razvoja algoritma je sestavljen iz krajše predstavitve mehke logike kateremu sledi podrobnejši opis mehkih spremenljivk in pripadajočih mehkih množic. Predstavljena so tudi pravila, ki so osnova za delovanje mehke logike ter izbor metode ostrenja. Določanje mehkih množic je podprto s strani rezultatov testiranja na samem sistemu.

Sledi primerjava rezultatov testiranja novega in starega algoritma. Izvedena je primerjava trajanja in primerjava odstopanja končne temperature od želene temperature pri segrevanju in vzdrževanju temperature. Prav tako so tu prikazani rezultati primerjave energijske porabe obeh algoritmov.

V zaključku so zbrane sklepne ugotovitve razvoja algoritma in predlogi za morebitno nadaljnje delo na tem področju.

Ključne besede: algoritem, mehka logika, segrevanje, vzdrževanje temperature

University of Ljubljana
Faculty of Computer and Information Science

Matjaž Okretič

**Fuzzy logic based heating and temperature maintenance
algorithm for the beer brewing process**

ABSTRACT

The thesis represents improved algorithm, which manages the number of active heaters for heating and maintaining temperature in the beer brewing process, based on fuzzy logic. The goals of the new algorithm are lower time and energy consumption in addition to lesser end temperature deviation from the desired temperature. Steps necessary for the algorithm development are divided into several chapters.

First is an environment presentation. It holds a description of kettle and its elements important for heating and maintaining the temperature. There is also a description of the whole beer brewing process and the existing algorithm of which results need to be improved. The tools used for the development are also presented in this chapter.

The second chapter is intended for determining the properties of the environment. Here we have the acquisition of environment data regarding the temperatures and active heater information in various probable scenarios as well as the result interpretation.

Algorithm development description consists of a short fuzzy logic presentation which is followed by an extensive presentation of fuzzy variables and associated fuzzy sets. The rules that act as a base for the fuzzy logic operation and the defuzzification method selection are also presented.

Next is the new versus old algorithm result comparison. It consists of duration comparison and a comparison of end temperature deviation from the desired temperature within heating and maintaining temperature. The energy consumption comparison of both algorithms is also presented.

Conclusion consists of algorithm development final findings and suggestions for possible further work in this area.

Key words: algorithm, fuzzy logic, heating, maintaining temperature

ZAHVALA

Zahvaljujem se mentorju doc. dr. Iztoku Lebarju Bajcu ter mag. Primožu Pečarju, za koristne nasvete in pomoč pri izdelavi diplomskega dela. Prav tako se zahvaljujem podjetju Batič d.o.o., za možnost uporabe njihovih prostorov in opreme na kateri so potekala testiranja.

Posebna zahvala pripada tudi celotni družini, za vso izkazano podporo, spodbudo in potrpežljivost v času mojega študija.

— Matjaž Okretič, Ljubljana, januar 2013.

KAZALO

Povzetek	i
Abstract	iii
Zahvala	v
1 Uvod	1
1.1 Motivacija in cilji	1
1.2 Struktura diplomskega dela	2
2 Okolje	3
2.1 Opis sistema	3
2.1.1 Kotel	3
2.2 Proces pridelave piva na sistemu	5
2.3 Uporabljena orodja	7
2.3.1 Beckhoff	7
2.3.2 Matlab	7
2.3.3 C#	8
2.4 Obstoječa rešitev segrevanja	8
3 Zajemanje podatkov	11
3.1 Opis zajemanja	11
3.2 Analiza	12
4 Razvoj	19
4.1 Lastnosti mehke logike	19
4.1.1 Mehke množice	19

4.1.2	Mehka logika	20
4.2	Algoritem	20
4.2.1	Spremenljivke	20
4.2.2	if-then pravila	24
4.2.3	Defuzifikacija	25
4.3	Razvoj algoritma	26
4.3.1	Spremenljivka P_1	27
4.3.2	Spremenljivka P_2	30
5	Rezultati	33
5.1	Primerjava rezultatov	33
6	Zaključek	39
6.1	Sklepne ugotovitve	39
6.2	Predlogi za nadaljnje delo	40

1 Uvod



1.1 Motivacija in cilji

Zgodovina piva in njenega varjenja se, po grobi oceni, začne na Kitajskem pred 6000 leti [1]. Od takrat se je enostavno varjenje pivine nadgradilo v zapleten proces kot ga poznamo danes. Ta proces vključuje varjenje, fermentiranje in hlajenje piva. Za optimalne rezultate se tako poslužujemo raznoraznih naprednih tehnologij s katerimi nadgrajujemo okolje varjenja piva.

V tem diplomskem delu vam bom predstavil problem in rešitev segrevanja vode in pivine na optimalno temperaturo, ter njeno vzdrževanje v danem sistemu. Motivacija tega dela izhaja iz dejstva, da je ključni del pri varjenju piva njen recept, ki vsebuje količinske, časovne in temperaturne informacije o procesu varjenja. Za konsistentno izvrševanje recepta pa je treba zagotoviti okolje, ki omogoča natančno sledenje njegovim zahtevam. Eno izmed področij okolja je tako segrevanje vode in pivine na točno določeno temperaturo in vzdrževanje te temperature za, v receptu, določen čas.

Cilji diplomskega dela so torej razvoj algoritma, ki bo na danem sistemu in pri danemu receptu, omogočil natančno sledenje želenim temperaturam in pa čim hitreje doseganje

želenih temperatur. Posredni cilj je tudi doseganje boljše energijske porabe pri procesih segrevanja in vzdrževanja v smislu zmanjšanja števila delujočih grelcev skozi posamezen proces.

1.2 Struktura diplomskega dela

Vsebinsko je diplomsko delo sestavljeno iz štirih delov. Prvi del predstavlja opis okolja v katerem se vrši varjenje. Predstavljena sta sistem in proces varjenja kot podlaga pri razvoju algoritma. Prvi del vsebuje še obstoječo rešitev segrevanja in vzdrževanja in opis orodij s katerimi sem si pri testiranju pomagal.

V drugem delu je opisan razvoj algoritma. Od začetnih faz zajemanja podatkov iz sistema in interpretacije dobljenih rezultatov, do predstavitve mehke logike in rešitve problema v njenem okolju.

V tretjem delu se nahaja še primerjava in analiza rezultatov obstoječe in nove rešitve.

V zadnjem delu pa so zbrane sklepne ugotovitve diplomskega dela. Podane so tudi možne in priporočljive izboljšave tako algoritma kot tudi okolja.

2 Okolje



2.1 Opis sistema

Sistem za varjenje piva je sestavljen iz dveh glavnih modulov vidnih na sliki 2.1. Zgornji del sistema je kotel namenjen kuhanju pivine, spodnji pa odcejanju prekuhanega ječmenovega slada in usedanju delcev hmelja pred prenosom v fermentatorje. Sistem ima tudi ekran na dotik, na katerem lahko spremljamo in spreminjamo stanje modulov (delovanje črpalke in ventilov, ročno vklapljanje grelcev, mešala,...).

Za segrevanje in vzdrževanje je pomemben le zgornji del sistema, kotel, saj se tu nahajajo grelci in senzorji. Pomemben del kotla je tudi mešalo, ki vodo oziroma pivino meša in tako temperaturo enakomerno porazdeli po masi in preprečuje vrenje vode ob stenah kotla.

2.1.1 Kotel

Slika 2.2 prikazuje skico kotla in njegovih elementov. Maksimalni volumen kotla je 572 litrov, ampak največja dovoljena količina vode je 550 litrov, saj moramo upoštevati še neenakomernost površine pri mešanju in tvorbo pene pri kuhanju. Količine potrebne



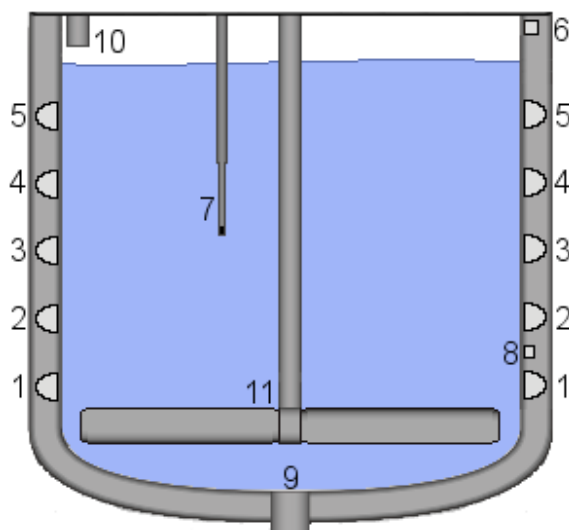
Slika 2.1 Sistem za varjenje piva.

za optimalni izkoristek sistema so opisane v poglavju 2.2. Kotel sestavljajo naslednji elementi:

1. *Grelci*: Sistem vsebuje pet električnih plaščnih grelcev, ki se nahajajo na zunanji strani notranje stene kotla (1-5 na sliki 2.2). Posamezen grelec deluje z močjo 5kW. Plašči grelcev so postavljeni v višinskem razmaku 15 centimetrov. Vsi grelci delujejo po ON-OFF principu, kar pomeni, da je posamezen grelec lahko le vklopljen ali izklopljen. Grelec pri dnu kotla je označen s številko 1, grelec na vrhu pa s številko 5. Tabela 2.1 prikazuje kateri grelci so vklopljeni v posameznem stanju.

Tabela 2.1 Stanje grelcev. Grelec 1 je na dnu kotla, grelec 5 pa na vrhu.

Število grelcev	Vklopljeni grelci	Število grelcev	Vklopljeni grelci
0	-	3	1, 2, 4
1	1	4	1, 2, 3, 4
2	1, 3	5	1, 2, 3, 4, 5



Slika 2.2 Skica elementov kotla; 1-5: grelci, 6: varnostni senzor, 7: senzor temperature medija, 8: senzor temperature stene, 9: odtok, 10: dotok, 11: mešalo.

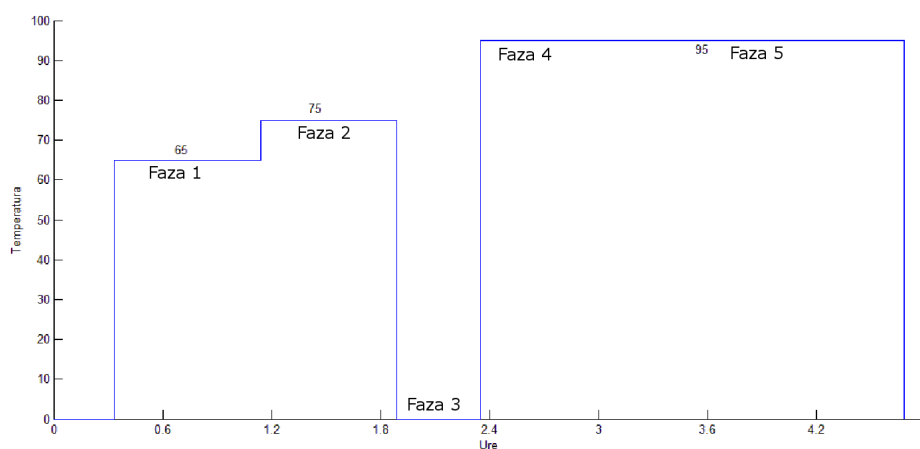
2. *Senzorja*: V kotlu se nahajata dva temperaturna senzorja (slika 2.2, 7 in 8). Prvi senzor služi za zajemanje temperature mase in se nahaja v bližini mešala. Drugi senzor je namenjen zajemanju temperature stene, ki jo tvorijo grelci. Ta je privarjen na notranjo stran stene med prvim in drugim spodnjim grelcem.
3. *Mešalo*: Mešalo se nahaja v sredini kotla in se vrti v smeri urinega kazalca (11 na sliki 2.2). Ima dve mešalni lopatici med seboj višinsko oddaljeni 30 centimetrov.
4. *Ostalo*: Elementi, ki niso povezani z razvojem algoritma (6,9,10 na sliki 2.2).

2.2 Proces pridelave piva na sistemu

Sistem je namenjen infuzijskemu načinu pridelave piva (drozganja), kar pomeni, da celotno sladno drozgo segrevamo postopoma. Slednje je različno od infuzijskega načina, pri katerem se segreva le del drozge in se jo nato vmeša v hladnejši del drozge [2]. Za zajemanje podatkov in testiranje sem uporabil splošen recept, ki se uporablja za varjenje piva na danem sistemu. Pri spodaj opisanem postopku je pomembna predvsem temperaturna komponenta skozi čas saj ostali deli recepta ne bistveno vplivajo na segrevanje medija. Približen temperaturni potek 5 urnega varjenja je predstavljen tudi v tabeli 2.2 in na sliki 2.3.

Tabela 2.2 Postopek pridelave piva iz vidika temperature.

Faza	Opis
1	Segrevanje na 65°C in 30 minutno vzdrževanje (dodajanje slada)
2	Segrevanje na 75°C in 30 minutno vzdrževanje (drozganje slada)
3	Tuširanje slada in ekstrakcija pivine (grelci ne delujejo)
4	Segrevanje na 95°C (dodajanje hmelja)
5	60 minutno vzdrževanje na 95°C



Slika 2.3 Temperaturni potek kuhanja.

Postopek varjenja približno 550 litrov pivine se začne z natakanjem 450 litrov vode v varilni kotel. Sledi segrevanje mase na 65°C. Ko se temperatura mase dvigne čez 60°C se v kotel doda večjo količino ječmenovega slada. Maso se pri tem ves čas meša. Ko masa doseže 65°C se začne 30 minutno vzdrževanje temperature. Sledi dvig zelene temperature mase na 75°C in nato spet 30 minutno vzdrževanje na tej temperaturi. Po končanem vzdrževanju se prekuhan slad pretoči na sito, kjer se filtrira pivina. Ob pretakanju so grelci izklopljeni. Ko se pivina izcedi iz slada, jo je potrebno pretočiti nazaj v varilni kotel. Na preostalem sladu se večkrat izvrši postopek tuširanja. Po vsakem tuširanju se nastala pivina pretoči v varilni kotel, dokler ni dosežena zelena količina pivine, v našem primeru 550 litrov. Ko je zelena količina pivine v varilnem kotlu se začne segrevanje na 95°C in naposled 60 minutno vzdrževanje.

Po zaključenem varjenju se pivino pretoči iz varilnega kotla v fermentator, ohladi v

okolico 17°C in doda kvas, ki začne proces fermentacije. Po fermentaciji sledi uležavanje in hlajenje v okolici 3°C.

2.3 Uporabljena orodja

Za zajemanje podatkov in razvoj algoritma sem uporabil okolje Matlab. Povezava Matlab in krmilnika, ki skrbi za upravljanje z grelci pa je realizirana s C# vmesnikom.

V realno-časovnem testiranju se algoritem izvaja v Matlab okolju, vhodni podatki in izhodne vrednosti algoritma se preko C# vmesnika prenašajo krmilnemu programu. Program krmilnika v tem načinu zaobide svoj osnovni algoritem za upravljanje z grelci in upošteva posredovane podatke, ki sporočijo kateri grelci delujejo v trenutnem ciklu. Zajemanje podatkov se vrši podobno, le da program krmilnika posreduje podatke C# vmesniku in ta jih preda Matlab okolju, ki skrbi tudi za celotno shranjevanje podatkov.

Izhodne vrednosti Matlab algoritma se posredujejo krmilniku ob vsaki spremembi. Če spremembe ni se ohranja prejšnje stanje cikla. Pri zajemanju podatkov je določen cikel dveh sekund, ob koncu katerega se iz krmilnika prebere podatkovna struktura, ki vsebuje podatke o temperaturnih senzorjih. Ti podatki se potem uporabijo pri izračunu novega števila delujočih grelcev.

2.3.1 Beckhoff

Sistem uporablja Beckhoff PLC krmilnik CX1100 na katerem deluje Windows Embedded sistem. Podatki o grelcih in temperaturah so shranjeni v podatkovni strukturi na krmilniku, ki se prebere s pomočjo C# vmesnika.

2.3.2 Matlab

Za razvoj algoritma sem uporabil Matlab orodje z imenom *Fuzzy Logic Toolbox*, ki je namenjeno delu z mehкими množicami. To orodje omogoča enostaven razvoj mehkih množic in pravil, prav tako pa je uporabno zaradi svojega vmesnika s katerim lahko opazujemo interakcijo mehkih množic in pripadajočih pravil. Več o samem izvajanju programa je opisano v poglavju 4.2. Dodatna funkcija, ki jo vrši Matlab, je zajemanje podatkov o času, želeni temperaturi, temperaturah na senzorjih in trenutno stanje grelcev. Podatki se shranjujejo v .csv (angl. *comma-separated values*) datoteko. Matlab algoritem uporablja shranjene podatke za sprotni prikaz grafov omenjenih spremenljivk. Prav tako so te datoteke uporabljene pri izdelavi grafov, ki so del tega diplomskega dela.

2.3.3 C#

Vmesnik izdelan v programskem jeziku C# uporablja dve knjižnici, ki sta nujni za komunikacijo med Matlab okoljem in Beckhoff krmilnikom. Knjižnica *MLApp* skrbi za povezavo z Matlab okoljem, tako da zgradi Matlab server na katerem se izvajajo Matlab ukazi. Knjižnica *TwinCat.Ads* pa je vmesnik med C# programom in ADS protokolom Beckhoff krmilnika, na kateremu se izvaja program za upravljanje z grelci.

Ko je nastavljena zelena temperatura vmesnik dobi izračun števila delujočih grelcev iz strani Matlab algoritma in ga posreduje Beckhoff krmilniku. Nato se vsaki dve sekundi prebere stanje temperaturnih senzorjev, ki se posreduje Matlab algoritmu. Matlab algoritem ponovno izvede izračun števila delujočih grelcev. Če se novo število delujočih grelcev razlikuje od prejšnjega, se novo število posreduje Beckhoff krmilniku. V nasprotnem primeru se stanje sistema ne spremeni.

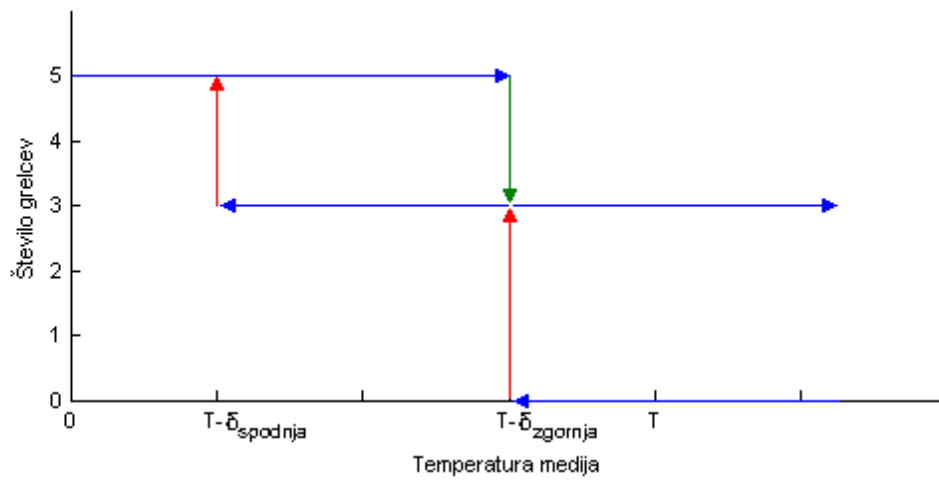
2.4 Obstoječa rešitev segrevanja

Obstoječi algoritem segrevanja je sestavljen iz dveh faz, faze segrevanja in faze vzdrževanja. Obe fazi delujeta na principu histerezne zanke, s čimer je preprečeno pretirano preklapljanje grelcev v primeru manjšega nihanja temperature. V prvi fazi je število delujočih grelcev odvisno od razmerja med temperaturo medija T_m in zeleno temperaturo T . Če je T_m manjša od $T - \delta_{spodnja}$ delujejo vsi grelci. Ko T_m doseže $T - \delta_{zgornja}$ se število delujočih grelcev spremeni na tri. Temperatura tako raste počasneje in je zato prekoračitev zelene temperature manjša. Faza vzdrževanja se začne, ko T_m doseže zeleno temperaturo. V tej točki delujeta le dva grelca, ki se histerezno vklopljata in izklopljata in tako zagotavljata konstantno vzdrževanje temperature. Vklon je pri $T + \delta_{vklop}$ izklon pa pri $T + \delta_{izklop}$. Obe fazi sta prikazani na slikah 2.4 in 2.5, vrednosti parametrov $\delta_{spodnja}$, $\delta_{zgornja}$, δ_{vklop} in δ_{izklop} pa so podane v tabeli 2.3.

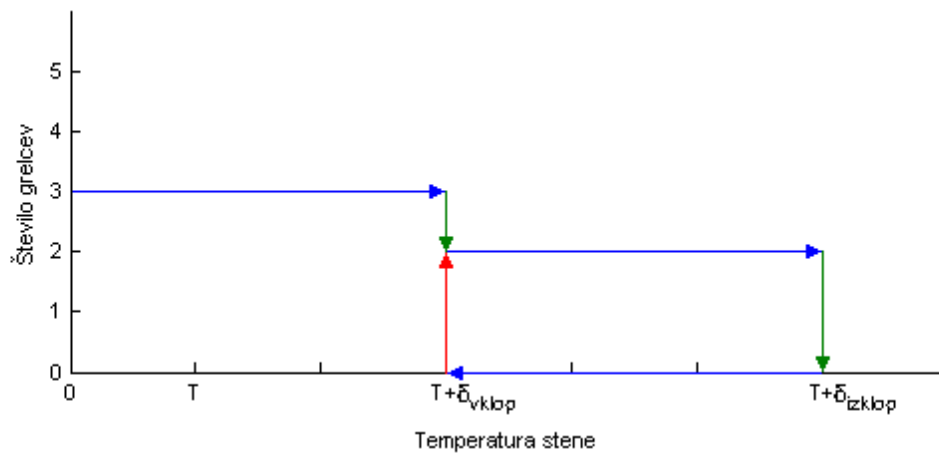
Zgoraj opisan algoritem služi tudi kot osnova pri načrtovanju mojega algoritma. Še posebej podobna je faza vzdrževanja, ki ravno tako vsebuje logiko na podlagi histerezne zanke.

Tabela 2.3 Vrednosti parametrov δ .

δ	Vrednost [$^{\circ}\text{C}$]
δ_{spodnja}	5
δ_{zgornja}	3
δ_{vklop}	2
δ_{izklop}	5



Slika 2.4 Graf faze segrevanja.



Slika 2.5 Graf faze vzdrževanja.

3 Zajemanje podatkov

3.1 Opis zajemanja

Za podrobnejše razumevanje poteka segrevanja je bilo potrebno najprej pridobiti podatke o samem delovanju sistema. Ker simulacija in modeliranje takega sistema predstavljata težko rešljiv problem, sem se odločil, da podatke pridobim na sistemu samem. Čeprav je takšno zajemanje časovno in energijsko potratno, so podatki pridobljeni na ta način natančni in odražajo osnovne lastnosti sistema.

Zajemanje podatkov sem razdelil na štiri dele. Najprej sem podatke zajemal pri posameznih vnaprej določenih temperaturah. Namen tega dela je bilo ugotavljanje morebitnih razlik v obnašanju sistema pri nižjih in višjih zelenih temperaturah ter pri večjih in manjših razlikah med temperaturo medija in želeno temperaturo.

Drugi del zajemanja je vseboval simulacijo procesa varjenja. Različne temperaturne razlike, ki se zvrstijo v tem procesu so mi bile v pomoč pri zajemanju podatkov postopnega segrevanja. Tako sem lahko približno ocenil obnašanje sistema brez pivine in ga nato primerjal z dejanskim procesom varjenja.

Tretji del zajemanja podatkov je bilo zajemanje sprememb temperatur pri dejanskem

procesu varjenja. Ta del je pomemben predvsem zaradi sestavin, ki spremenijo gostoto mase v kotlu in posledično tudi obnašanje celotnega sistema. Za zajemanje podatkov sem uporabil obstoječi algoritem segrevanja. V nadaljevanju je predstavljena analiza pridobljenih podatkov.

Zadnji del pa predstavlja zajemanje podatkov o spremembi temperature v določenem časovnem obdobju pri različnem številu delujočih grelcev. Ta del je pomemben predvsem za fazo vzdrževanja, saj je iz njega razvidno kakšna je optimalna obremenitev grelcev za doseganje minimalne spremembe temperature pri njenem vzdrževanju.

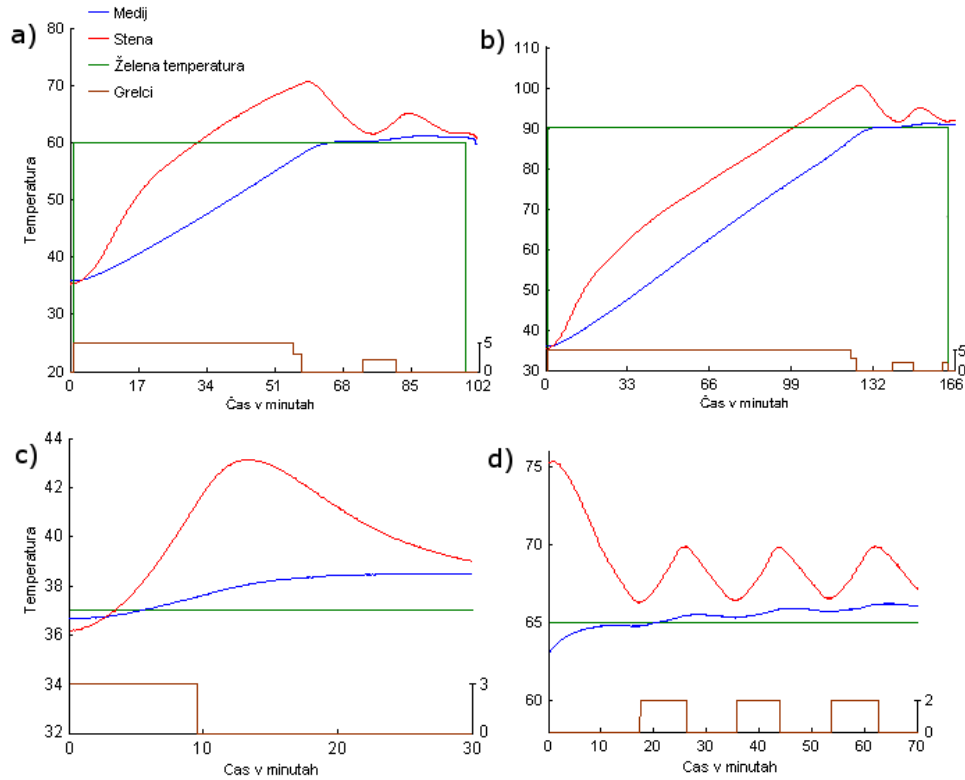
3.2 Analiza

Slika 3.1 prikazuje potek segrevanja in 30 minutnega vzdrževanja z obstoječim algoritmom pri temperaturi 60°C (a) in 90°C (b) iz približno iste začetne temperature ter obnašanje sistema če je zelena temperatura v okolici začetne (c). Na grafu (d) pa je prikazan potek eno urnega vzdrževanja.

Iz slike 3.2 je razvidno, da se razlika med temperaturo stene in temperaturo mase pri segrevanju na različne temperature ne bistveno spreminja. V okolici zelene temperature temperatura stene premočno naraste in povzroči večje odstopanje temperature medija od zelene temperature. Problematičen del je predvsem lovljenje zelene temperature, ki mora biti časovno optimalen glede na zeleno minimalno napako.

Segrevanje do okolice zelene temperature ne predstavlja problema, saj so v tem primeru maksimalno obremenjeni vsi grelci sistema. Temperatura medija doseže vrednost 60°C po 62 minutah, vrednost 90°C pa po 132 minutah. Vseh pet grelcev deluje dokler ni razlika med zeleno temperaturo in temperaturo medija manjša od 3°C. Maksimalna vrednost temperature stene preseže zeleno temperaturno vrednost 60°C za približno 11°C, 90°C pa za nekaj več kot 10°C. Maksimalna razlika med temperaturo medija in temperaturo stene je pri segrevanju na 60°C v okolici 14°C, pri segrevanju na 90°C pa približno 15.5°C.

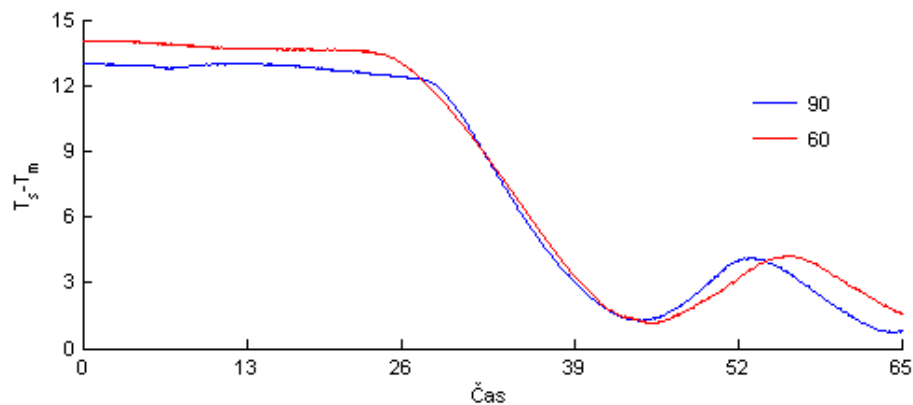
Graf 3.1c prikazuje segrevanje, ko je začetna temperatura medija T_m v okolici končne zelene temperature T . V danem primeru je razlika med začetno temperaturo stene in zeleno temperaturo 0.8°C in razlika med začetno temperaturo medija in zeleno temperaturo 0.3°C. Kljub majhni razliki temperatura stene močno naraste, saj so vključeni trije grelci, ki jih izklopi šele vzdrževalni dela algoritma, ko temperatura stene doseže



Slika 3.1 a) segrevanje in vzdrževanje na 60°C, b) segrevanje in vzdrževanje na 90°C, c) segrevanje, ko je začetna temperatura v okolici zelene temperature, d) vzdrževanje na 65°C.

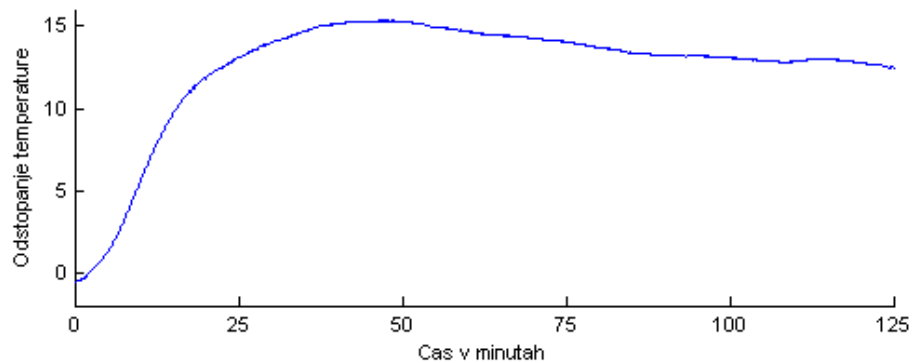
$T + \delta_{izklop}$ (glej poglavje 2.4). Zaradi tega temperatura medija narašča tudi, ko je zelena temperatura že dosežena.

Pri vzdrževanju obstoječi algoritem ne zagotavlja konstantnega minimalnega odstopanja od zelene temperature, saj se temperatura mase s časom viša in tako povečuje napako, ki mora biti v tem primeru čim manjša. Večanje napake je posledica prevelike razlike med temperaturo stene in zeleno temperaturo. Ko temperatura medija doseže zeleno temperaturo 60°C je temperatura stene od te višja za 7.5°C. V naslednjih 10 minutah razlika med temperaturo stene in zeleno temperaturo pade na 1.5°C, pri tem temperaturo medija dvigne za 0.25°C, kar je še v mejah napake. Vendar se nato temperatura stene začne spet dvigati do razlike 5°C, kar traja približno 10 minut, in pri tem dvigne temperaturo medija na 61°C kar je na meji napake. Pri daljšem vzdrževanju se ta cikel ponovi in tako temperatura medija naraste čez zeleno mejo napake 1°C, kar je



Slika 3.2 Razlika med temperaturo stene in temperaturo mase pri segrevanju na 60°C in 90°C od trenutka, ko temperatura stene doseže želeno temperaturo.

razvidno tudi na grafu 3.1d. Podobno je pri segrevanju na 90°C. Ob doseženi željeni temperaturi je temperatura stene višja za 7.3°C, ko temperatura stene pade na razliko 1.5°C je temperatura medija višja za 0.26°C in naraste čez mejo 91°C, ko se temperatura stene spet zviša. V tretjem ciklu je napaka že višja od 1.5°C in če bi se vzdrževanje nadaljevalo bi temperatura medija limitirala proti 5°C napake. Slednje zaradi algoritma, ki v tem primeru gleda le razliko stene in želene temperature.

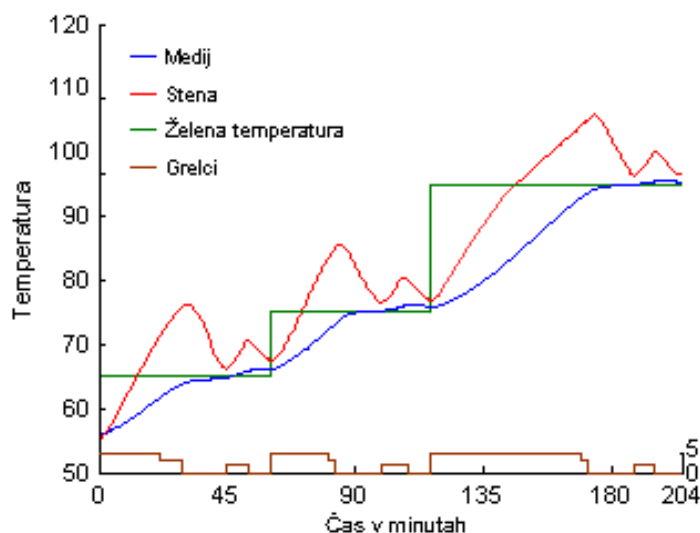


Slika 3.3 Odstopanje med temperaturo stene in temperaturo medija pri segrevanju na 90 °C, ko deluje vseh pet grelcev.

Iz grafa 3.3 je razvidno, da se temperatura stene ne vedno oddaljuje od temperature medija, ampak se po določeni točki začne približevati, kljub temu da ves čas deluje vseh pet grelcev. Maksimalno razliko 15.5°C doseže po 47 minutah, nato pa se začne razlika

manjšati vse do 12.5°C , kljub temu da deluje vseh pet grelcev.

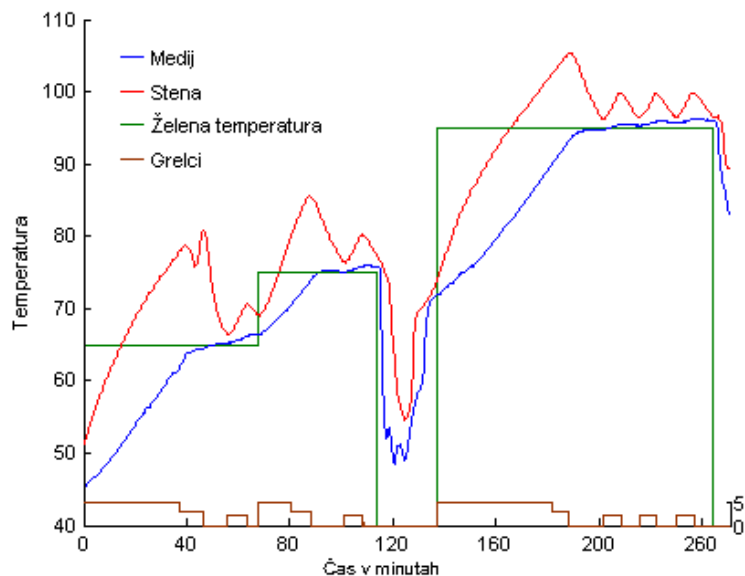
Iz prikazanih grafov in razlag lahko ugotovimo, da se segrevani medij podobno obnaša če je želena temperatura v okolici začetne, če je razlika med začetno in želeno temperaturo v okolici 20°C in če je razlika med začetno in želeno temperaturo v okolici 50°C . To pomeni, da želena temperatura ne vpliva na delovanje algoritma tako v fazi segrevanja kot tudi vzdrževanja. Pri segrevanju, kjer je začetna temperatura medija v okolici želene temperature, je potrebno zagotoviti, da se po doseženi želeni temperaturi temperatura medija ne dviga. Slednje ugotovitve so osnova za izgradnjo novega algoritma, ki poskuša odpraviti veliko odstopanje temperature stene v okolici želene temperature in odpravlja večanje napake pri daljšem vzdrževanju.



Slika 3.4 Graf simulacije različnih temperatur.

Slika 3.4 prikazuje potek temperatur pri simulaciji varjenja, začetna temperatura je približno 56°C . V prvih 31 minutah se voda segreje na 65°C , sledi 30 minutno vzdrževanje. Ob koncu prvega dela je napaka vzdrževanja ravno na meji 1°C . Nato se želena temperatura poveča na 75°C . Za doseganje nove temperature algoritem porabi nekaj manj kot 30 minut segrevanja, napaka po 30 minutnem vzdrževanju je tudi tukaj ravno na meji dovoljenega. Za zvišanje temperature iz 75°C na 90°C je potrebno počakati približno 65 minut in nato sledi še 30 minutno vzdrževanje. Iz dobljenega poteka je razvidno, da se lahko ugotovitve iz zajemanja podatkov za posamezne temperature prenesejo tudi na

proces varjenja.



Slika 3.5 Graf dejanskega poteka kuhanja.

Tudi pri dejanskem procesu varjenja so ugotovitve iste (slika 3.5). Zanimivo je dejstvo, da dodajanje sestavin ne vpliva na temperaturo medija, kot je bilo to pričakovati. Iz slike je razvidno, da je edina zaznavna sprememba temperature stene okoli štiridesete minute, ko se dodaja slad. Očiten padec in sledeča previsoka rast temperature stene je v veliki meri odvisna tudi od mešanja sestavin v medij. Če se prepusti mešanje samo mešalu v kotlu je skok temperature veliko večji kot če se ob vnosu sestavin izvaja še ročno mešanje, da se slad ne nabira ob steni kotla. Nov algoritem mora biti odporen na takšna popačenja temperature stene. Ena izmed razlik med simulacijo kuhanja in dejanskim kuhanjem je tudi predel, kjer se vrši tuširanje. Na grafu iz slike 3.5 je lepo razvidno kako temperatura v kotlu pade, ko v njem ni medija. Ob ponovnem natakanju v kotel pa temperatura spet poskoči, ko medij zaliže pozicije temperaturnih senzorjev. Tudi na tem grafu je razvidna napaka, ki je bila omenjena že pri testiranju vzdrževanja na daljše obdobje. In sicer po 60 minutnem vzdrževanju na 95°C je temperatura medija že 96.5°C, kar pomeni 1.5°C napake. Želena maksimalna napaka algoritma pa je 1°C.

Iz analize sledi, da je prostor za izboljšavo algoritma predvsem pri lovljenju zelene temperature tako, da bo dosežena minimalna napaka v najkrajšem možnem času. Prav

tako je potrebno izboljšanje algoritma pri vzdrževanju zelene temperature, saj se v obstoječem primeru temperatura konstantno viša in po daljšem času zapusti dovoljeno območje najmanjše napake.

Za boljše rezultate pri vzdrževanju, sem opravil še primerjavo segrevanja z različnim številom delujočih grelcev. Zanimala me je predvsem optimalna odzivnost sistema v kombinaciji s čim manjšim energijskim vložkom. Na sliki 3.6 je prikazanih vseh pet grafov segrevanja v 15 minutnem časovnem intervalu.

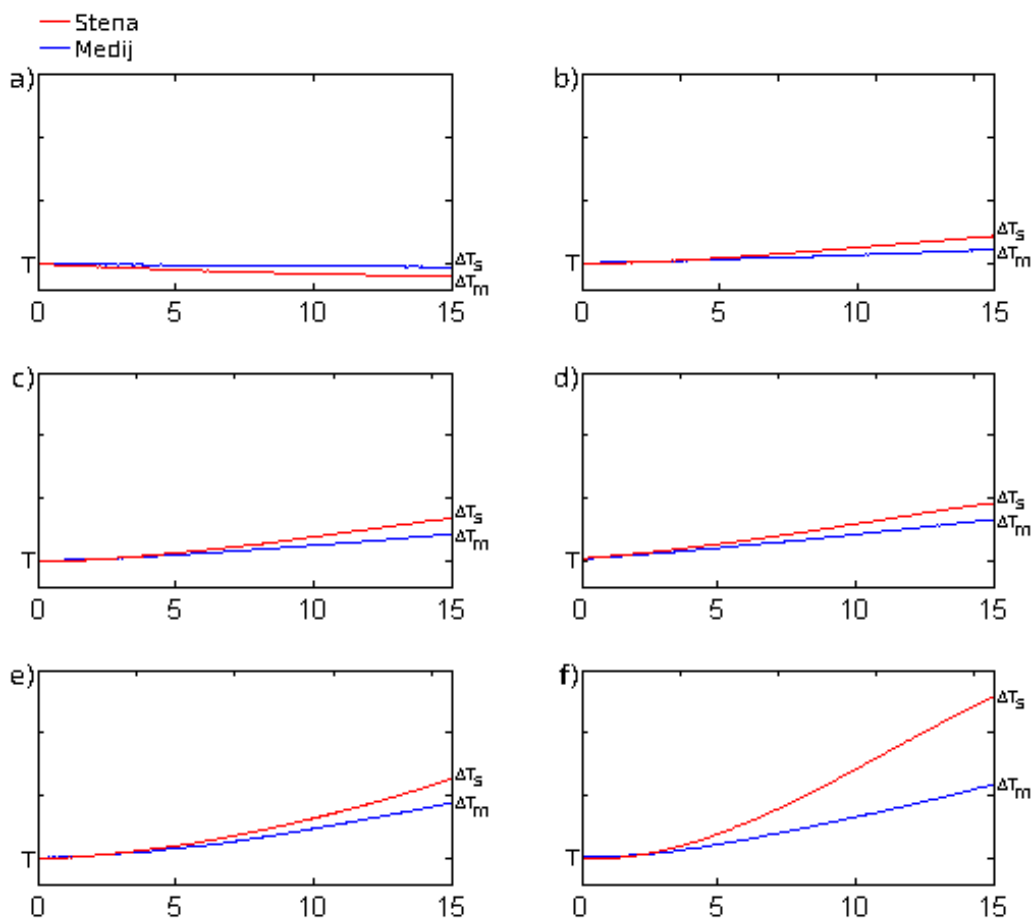
Rezultati iz tabele 3.1 in grafov na sliki 3.6 prikazujejo spremembo temperature pri različnem številu delujočih grelcev sistema v 15 minutnem intervalu segrevanja. Za konsistentnost rezultatov je bilo segrevanje v vseh primerih izvršeno iz približno iste začetne temperature medija, 40°C. Grelci pričnejo z delovanjem, ko temperatura stene T_s pade pod temperaturo medija T_m za 0.1°C. Kot že omenjeno merjenje traja 15 minut, nakar se grelci izklopijo.

Tabela 3.1 Tabela sprememb temperature posameznega števila vključenih grelcev v 15 minutnem intervalu.

Število grelcev	ΔT_m (°C)	ΔT_s (°C)
0	-0.3	-0.6
1	1.1	2.2
2	2.1	3.5
3	3.2	4.9
4	3.9	6.1
5	5.5	12.9

V primeru na sliki 3.6 graf *a*), ko ne deluje noben grelec, padata tako temperatura stene, kot tudi temperatura medija. Temperatura stene T_s pada hitreje kot temperatura medija T_m , tako da slednja ne mora biti nižja v nobenem primeru. Ker je začetna T_s od začetne T_m manjša za 0.1°C, moramo za pridobitev razlike med temperaturama vrednostim v stolpcu ΔT_{stene} iz tabele 3.1 odšteti 0.1°C. Sledi da je razlika med T_s in T_m v primeru *b* 1°C, v primeru *c* 1.3°C. Če delujejo trije grelci (primer *d*) je razlika 1.6°C. Če pa delujejo štirje ali pet grelcev pa je razlika 2°C oz. 7.3°C.

Kot je bilo pričakovati narašča tudi razlika med temperaturama v odvisnosti od števila delujočih grelcev. Ker vzdrževanje rabi natančne premike oziroma popravke temperature, je potrebno izbrati takšno število grelcev, ki v najkrajšem času izvede spremembo zelene temperature, vendar pri tem ne povzroči prevelike inercije sistema. Na podlagi



Slika 3.6 Grafi 15 minutnega segrevanja z različnim številom grelcev; a) brez grelcev, b) en grelec, c) dva grelca, d) trije grelci, e) štiri grelci, f) pet grelcev.

povedanega, sem za proces vzdrževanja izbral kot osnovo primer, ko delujejo trije grelci.

Pred začetkom razvoja algoritma, je bilo potrebno opredeliti še maksimalno dovoljeno odstopanje temperature medija T_m od želene temperature T . Optimalno delovanje bi omogočala napaka do 0.5°C , pri čemer je bolj pomembno, da je želena temperatura presežena, kot pa da je T_m v območju $T - 0.5^\circ\text{C}$. Zato sem se odločil, da mora algoritem zagotoviti maksimalno odstopanje 1°C , pri čemer mora biti zagotovljeno, da bo želena temperatura presežena. Optimizacija napake je nujna in bolj pomembna od časovne komponente segrevanja.

4 Razvoj



4.1 Lastnosti mehke logike

Algoritem opisan v poglavju 4.2 deluje na osnovi mehke logike (angl. *fuzzy logic*). S pomočjo mehkih množic in pripadajočih pravil se tvori število delujočih grecev v danem trenutku. Sledi kratek opis konstruktorov potrebnih za izgradnjo algoritma v mehki logiki.

4.1.1 Mehke množice

Mehke množice (angl. *fuzzy sets*) predstavljajo intuitivno nadgradnjo in razširitev klasičnih množic (angl. *crisp sets*). Klasična množica je lahko definirana s točno določenimi elementi ali z deklaracijo natančnih pravil, ki določajo pripadnost elementov. Pripadnostna funkcija (angl. *membership function*) take množice preslika elemente v množico $\{1,0\}$. Preslikava v 1 se vrši, če element zadovoljuje pogoje pripadnostne funkcije (torej pripada množici) in preslikava v 0, če jih ne (ne pripada množici) [3].

Razširitev klasičnih množic predstavljajo mehke množice. Take množice vsebujejo elemente, ki nimajo točno določenih lastnosti. Element lahko množici pripada le delno. Pripadnostna funkcija mehke množice določa stopnjo pripadnosti (angl. *degree of mem-*

bership) elementa tej množici. Tako dobimo preslikavo v realno število na intervalu $[0, 1]$. Posamezen element lahko tako množici pripada le delno [3].

4.1.2 Mehka logika

S pomočjo operatorjev *and*, *or* in *not* lahko iz enostavnih trditev nad mehкими množicami tvorimo kompleksnejše. V tem primeru se najprej ugotovi resničnost enostavne trditve, nato pa s pomočjo operatorjev še resničnost kompleksnejše trditve.

V mehki logiki je pogosta tudi uporaba *if-then* pravila v obliki '**if** A **then** C ' pri čemer sta A in C trditvi. Vhodne trditve (*if*) lahko med seboj povežemo z *and* operatorjem in tako dobimo množico pogojev, ki morajo biti izpolnjeni za določeno izhodno trditve (*then*). Prav tako lahko vse vhodne trditve z isto izhodno trditvijo povežemo z operatorjem *or* in interpretiramo kot eno samo *if-then* pravilo.

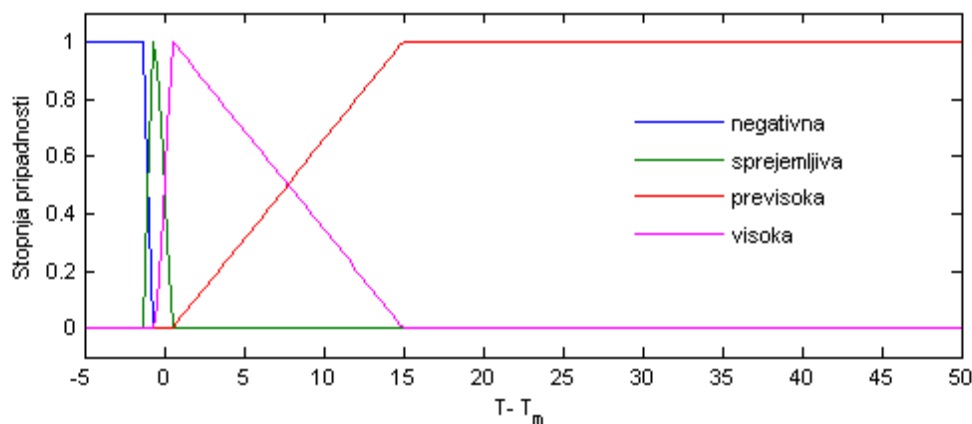
Pogosto se zgodi, da se v odločitvenem procesu *if-then* pravil tvori več izhodnih množic, ki jih nato združimo v eno samo množico. V primeru da proces potrebuje za delovanje ostro vrednost, je potrebno množico izhodnih vrednosti pretvoriti v eno samo ostro vrednost s pomočjo ostrenja (angl. *defuzzification*). Proces ostrenja iz izhodne množice izbere najbolj reprezentativno ostro vrednost na podlagi izbrane metode ostrenja. Najpogostejša metoda ostrenja je težiščna metoda (angl. *center of gravity*), ki kot rezultat vrne težišče pripadnostnih funkcij unije izhodnih množic, ki so posledica *if-then* pravil.

4.2 Algoritem

Za razvoj algoritma sem uporabil ugotovitve in rezultate iz poglavja 3.2. Pri izgradnji pripadnostnih funkcij je bilo potrebno zagotoviti učinkovito delovanje algoritma tako pri segrevanju, kot tudi pri vzdrževanju temperature. Po postopnem nadgrajevanju algoritma so kot končni produkt nastale tri vhodne in ena izhodna spremenljivka. V nadaljevanju so najprej opisane mehke spremenljivke, nato pa še njihova medsebojna interakcija v obliki *if-then* pravil. Na koncu je še opis uporabljene metode ostrenja.

4.2.1 Spremenljivke

Vhodna spremenljivka P_1 , ki je prikazana na sliki 4.1, je razbita na štiri mehke množice. Spremenljivko opredeljuje razlika med zeleno temperaturo T in trenutno temperaturo



Slika 4.1 Pripadnostne funkcije mehkih množic spremenljivke P_1 .

medija t_m v $^{\circ}\text{C}$.

$$P_1 = T - T_m \quad (4.1)$$

Množica *previsoka* ima skupaj z množico *visoka* glavno vlogo pri segrevanju do želene temperature. Množica *visoka* pa sodeluje tudi pri vzdrževanju temperature in sicer skupaj z množicama *sprejemljiva* in *negativna*. Mehke množice *negativna*, *sprejemljiva* in *previsoka* so trapezne oblike, množica *visoka* pa trikotne. Definicije mehkih množic spremenljivke P_1 so:

$$\textit{negativna} = (-5, -5, -1.25, -0.75) \quad (4.2)$$

$$\textit{sprejemljiva} = (-1.25, -0.75, -0.5, 0.5) \quad (4.3)$$

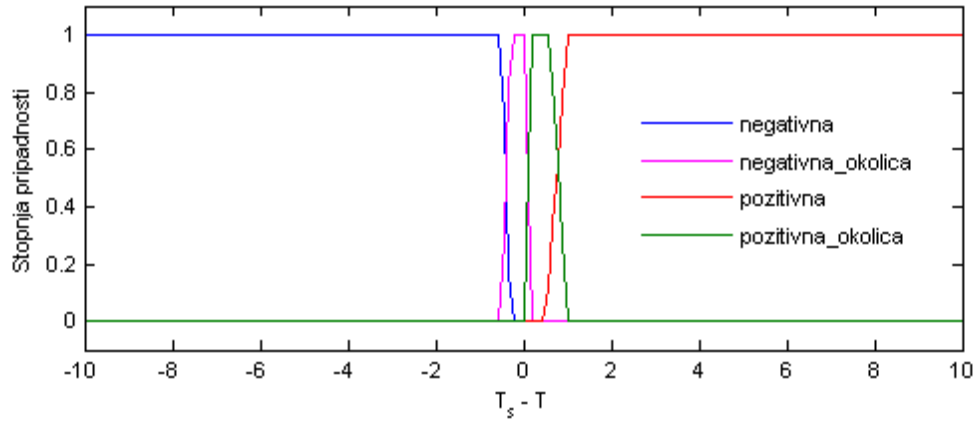
$$\textit{visoka} = (-0.5, 0.5, 15) \quad (4.4)$$

$$\textit{previsoka} = (0.5, 15, 50, 50) \quad (4.5)$$

Meje množic so izbrane ob predpostavki, da je razlika iz enačbe (4.1) vedno manjša od 50°C . Če je razlika manjša od -5°C pa ne vpliva na delovanje algoritma, zato ker upravljanje prevzame vrednost spremenljivke P_2 . Ostale vrednosti so izbrane glede na testna segrevanja opisana v poglavju 4.3.

Tudi spremenljivka P_2 , prikazana na sliki 4.2, vsebuje štiri mehke množice trapezne oblike, katerih elementi pripadajo razliki med želeno temperaturo T in trenutno temperaturo stene T_s v $^{\circ}\text{C}$:

$$P_2 = T_s - T \quad (4.6)$$



Slika 4.2 Pripadnostne funkcije mehkih množic spremenljivke P_2 .

Definicije mehkih množic spremenljivke P_2 :

$$\textit{negativna} = (-10, -10, -0.5, -0.3) \quad (4.7)$$

$$\textit{negativna_okolica} = (-0.5, -0.3, 0, 0.2) \quad (4.8)$$

$$\textit{pozitivna_okolica} = (0, 0.2, 0.6, 1) \quad (4.9)$$

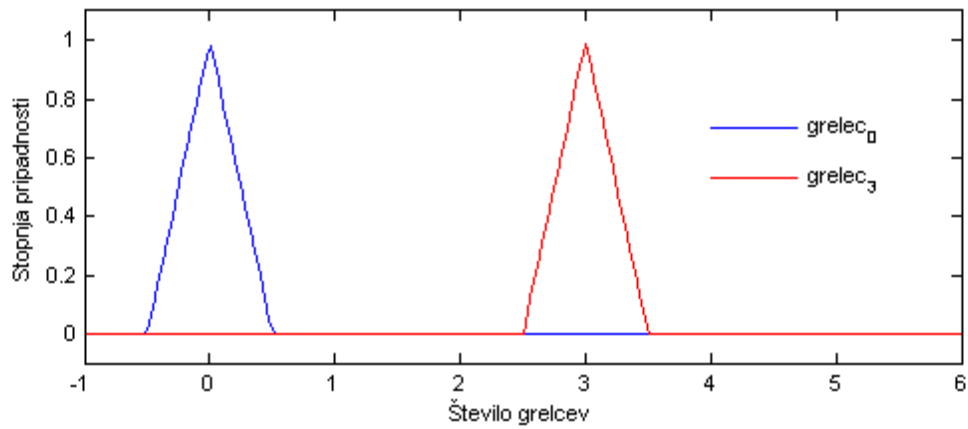
$$\textit{pozitivna} = (0.6, 1, 10, 10) \quad (4.10)$$

Ta spremenljivka vpliva na izhodno funkcijo le v območju vzdrževanja temperature. Več o interakciji spremenljivk je opisano v nadaljevanju pri predstavitvi *if-then* pravil. Tudi pri tej funkciji je bilo potrebno določiti optimalne vrednosti mehkih množic. Dobljene definicije so rezultat testiranja opisanih v poglavju 4.3.

Spremenljivka na sliki 4.3 je P_3 . To je vrednost prejšnjega stanja, ki je potrebna za izvedbo histereznega preklapljanja pri vzdrževanju temperature. Vsebuje dve trikotni mehki množici. Ker je pri vzdrževanju možen le izklop grelcev ali vklop treh grelcev, možna vhodna vrednost spremenljivke P_3 pripada množici 0, 3. Ker je vhod celo število bi lahko te množice definirali tudi kot običajne, saj je pripadnost vhodnega elementa vedno 1. Definicija mehkih množic spremenljivke P_3 :

$$\textit{grelec}_0 = (-0.5, 0, 0.5) \quad (4.11)$$

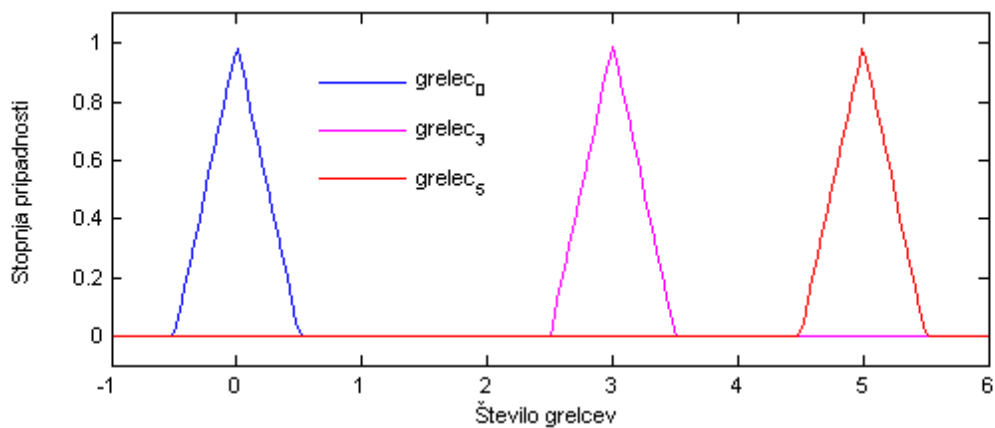
$$\textit{grelec}_3 = (2.5, 3, 3.5) \quad (4.12)$$



Slika 4.3 Pripadnostne funkcije mehkih množic spremenljivke P3.

Izhodna spremenljivka O (slika 4.4) vsebuje tri možna stanja grelcev glede na pravila iz tabele 4.1; Vsi grelci so izklopljeni, delujejo trije grelci in deluje vseh pet grelcev. Čeprav izbrana postavitev mehkih množic ne izkorišča glavnih lastnosti mehke logike, sem se odločil, da bom množice ohranil v opisani obliki. Definicija mehkih množic spremenljivke O :

$$grelec_n = (n - 0.5, n, n + 0.5), n \in \{0, 3, 5\} \quad (4.13)$$



Slika 4.4 Pripadnostne funkcije mehkih množic spremenljivke O.

4.2.2 if-then pravila

Tabela 4.1 vsebuje *if-then* pravila, ki določajo izhodno spremenljivko O . Pravili 1 in 2 sta osnovni pravili, ki skrbita za robne pogoje. V primeru, ko je P_1 v območju mehke množice *previsoka* je potrebno segrevanje. Vklopi se vseh 5 grelcev oziroma vsi grelci, ki so v danem trenutku na voljo (če je kateri izmed grelcev v okvari potem je izklopljen iz sistema). Če pa je P_1 v območju mehke množice negativna sledi izklop grelcev, za katerim sledi še segrevanje zaradi inercije sistema. Ščasoma temperatura stene preneha rasti. Omenjeni dve pravili sta tudi edini dve, ki nista del histerezne zanke.

Tabela 4.1 *if-then* pravila, ki določajo izhodno spremenljivko O .

Št.	<i>if</i> P_1 is	<i>and</i> P_2 is	<i>and</i> P_3 is	<i>then</i> O is
1	<i>negativna</i>	-	-	<i>grelec₀</i>
2	<i>previsoka</i>	-	-	<i>grelec₅</i>
3	<i>visoka</i>	<i>negativna</i>	-	<i>grelec₃</i>
4	<i>visoka</i>	<i>pozitivna</i>	-	<i>grelec₀</i>
5	<i>visoka</i>	<i>negativna_okolica</i>	-	<i>grelec₃</i>
6	<i>visoka</i>	<i>pozitivna_okolica</i>	<i>grelec₀</i>	<i>grelec₀</i>
7	<i>visoka</i>	<i>pozitivna_okolica</i>	<i>grelec₃</i>	<i>grelec₃</i>
8	<i>sprejemljiva</i>	<i>negativna</i>	-	<i>grelec₃</i>
9	<i>sprejemljiva</i>	<i>pozitivna</i>	-	<i>grelec₀</i>
10	<i>sprejemljiva</i>	<i>pozitivna_okolica</i>	-	<i>grelec₀</i>
11	<i>sprejemljiva</i>	<i>negativna_okolica</i>	<i>grelec₀</i>	<i>grelec₀</i>
12	<i>sprejemljiva</i>	<i>negativna_okolica</i>	<i>grelec₃</i>	<i>grelec₃</i>

Ko P_1 vstopi v območje mehke množice *visoka* obstaja več različnih scenarijev za delovanje grelcev. Če je P_2 v območju mehke množice *negativna* (pravilo 3) ali *negativna_okolica* (pravilo 5) nadaljujemo segrevanje s tremi grelci, saj je temperatura medija že zelo blizu zeleni temperaturi in je potrebno segrevanje zmanjšati. V primeru, da segrevanja ne bi zmanjšali, bi se temperatura stene preveč oddaljila od zelene temperature in bi temperatura medija prekoračila dovoljeno odstopanje od zelene temperature, ki je 1°C. Ker trije grelci zagotavljajo dobro razmerje med dvigovanjem temperature stene in temperature medija, so tudi v tem primeru dobra izbira. Pravilo 4 izklopi vse grelce, saj je v tem primeru dosežena razlika P_2 previsoka za območje *visoka* v spremenljivki P_1 . Če je P_2 v

območju *negativna_okolica* izhodna spremenljivka O predstavlja prejšnje stanje sistema. Pravila 3-7 tvorijo histerezno zanko mehke množice *visoko*.

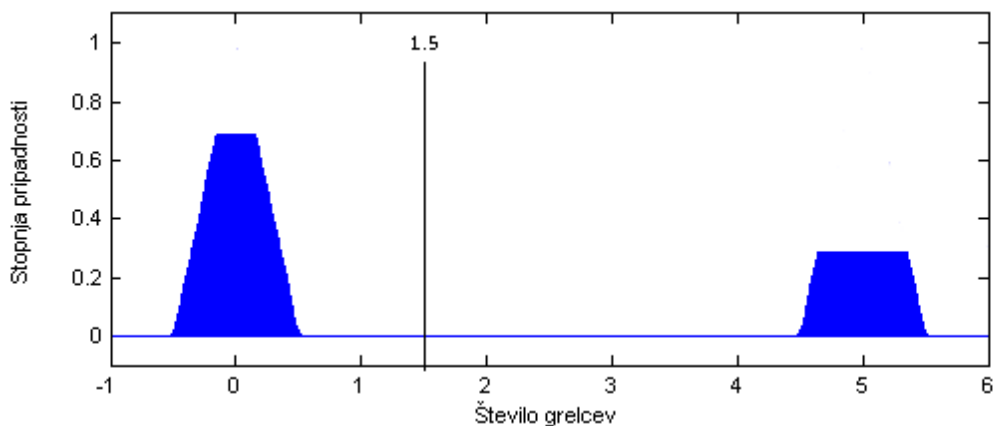
V primeru, ko je P_1 v območju mehke množice *sprejemljiva*, gre za vzdrževanje dosežene temperature. To je histerezna zanka katere pogoja za spreminjanje sta pravili 8 in 10. Ko je razlika P_2 v območju mehke množice *negativna* pomeni, da je potrebno pri vzdrževanju dvigniti temperaturo in se zato vključijo trije grelci. Trije grelci so ponovno uporabljeni zato, ker predstavljajo kompromis med hitrostjo dviga temperature, porabo energije in obrabo elementa. Upoštevati je treba tudi inercijo sistema, kar pomeni, da stena oddaja svojo temperaturo v medij tudi, ko so grelci že izklopljeni. Trije grelci zagotavljajo maksimalno dovoljeno inercijo sistema. Če bi uporabili manj grelcev bi bila inercija sistema manjša, a bi bil interval segrevanja daljši in s tem podaljšana obremenitev izbranih grelcev. Če je razlika P_2 v območju *pozitivna_okolica* pa je potrebno grelce izklopiti, da temperatura medija ne naraste preveč. S pravili 10-12 se zagotavlja ohranjanje prejšnjega stanja sistema za potrebe histerezne zanke. Ko je P_2 v območju *negativna_okolica* je vrednost izhodne spremenljivke prejšnje stanje sistema. Pravilo 10 je le prehodno pravilo, ko P_1 prehaja iz mehke množice *visoko* v mehko množico *sprejemljivo*.

4.2.3 Defuzifikacija

Defuzifikacija ali ostrenje poteka po metodi COG (angl. *Center of gravity*). Glavno vlogo ima pri prehodu iz mehke množice *previsoko* v mehko množico *visoko* spremenljivke P_1 . Temperaturo stene je potrebno primerno znižati, da temperatura medija ne preseže zelene temperature za več kot je to dovoljeno. Pri samem vzdrževanju je defuzifikacija enostavna, saj gre za histerezni zanki, pri katerih se le posreduje prejšnje stanje sistema in je tako vedno izbrana le ena mehka množica izhodne spremenljivke 0.

Za primer vzemimo, da je razlika med zeleno temperaturo T in temperaturo medija T_m enaka 4.85°C , razlika med temperaturo stene T_s zeleno temperaturo T pa 3°C . V prejšnjem stanju sistema so delovali trije grelci. Stopnje pripadnosti mehkih množic vhodnih spremenljivk so $P_1(4.85) = \{0, 0, 0.7, 0.3\}$, $P_2(3) = \{0, 0, 0, 1\}$ in $P_3(3) = \{0, 1\}$. Ob podanih podatkih, na rezultat vplivata dve pravili iz tabele 4.1. Pravilo 2 mehki množici *grelc₅* izhodne spremenljivke poda stopnjo pripadnosti 0.3, s pravilom 10 pa je pripadnost mehki množici *grelc₀* izhodne spremenljivke enaka 0.7. Vrednost vhodne spremenljivke P_3 v tem primeru ne vpliva na izhodno vrednost. Pripadnost mehkih

množic izhodne spremenljivke je torej $O(4.5, 3, 3) = \{0.7, 0, 0.3\}$. Ob upoštevanju COG metode ostrenja, dobimo izhodno vrednost 1.5. Po pravilu zaokroževanja morata v danem primeru delovati dva grelca. Primer je prikazan tudi na sliki 4.5.



Slika 4.5 Interakcija *if-then* pravil in defuzifikacija.

4.3 Razvoj algoritma

Za določitev vrednosti mehkih množic spremenljivk P_1 in P_2 je bilo potrebno izvesti serijo testiranj, z različnimi postavitvami mehkih množic in naposled izbrati množice, ki pri segrevanju in vzdrževanju dajo najboljše rezultate. Prvi del testiranja se osredotoča na prehod iz segrevanja na vzdrževanje. Gre za obnašanje v okolici želene temperature s strani segrevanja. Za potrebe prvega dela je bilo potrebno uvesti še filtriranje temperature, da se izognemo morebitnim preklpom grelcev med samim segrevanjem. Filter je s formulo (4.14) podrobneje opisan na koncu poglavja. Drugi del testiranja je delovanje grelcev pri vzdrževanju. Zaradi inercije sistema je potrebno določiti minimalne vrednosti mehkih množic, ki ohranjajo temperaturo medija v zelenih okvirih.

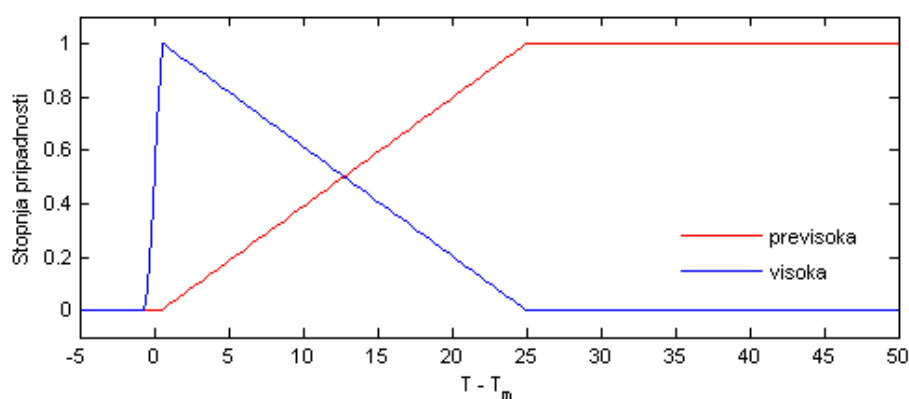
Spremenljivki P_3 in O sta izbrani na podlagi rezultatov testiranj iz katerih je razvidno, da algoritem za svoje delovanje potrebuje le tri stanja. Stanje, ko so grelci izklopljeni, ko delujejo trije grelci in ko je vklopljenih vseh pet grelcev. Mehka logika pa poskrbi, da se na podlagi izbranih pravil določijo vmesna stanja grelcev in tako optimizira prehod temperature.

4.3.1 Spremenljivka P_1

Mehka spremenljivka P_1 ponazarja razliko želene temperature in temperature medija $T - T_m$ ter vsebuje štiri mehke množice oziroma območja delovanja. Množica *previsoka* skrbi za segrevanje, ko temperatura medija še ni v območju želene temperature. Množica *visoka* je prehod iz segrevanja na vzdrževanje. V območju množice *sprejemljiva* pa poteka vzdrževanje. Množica *negativna* grelce izklopi ob morebitnem prekoračenju temperature.

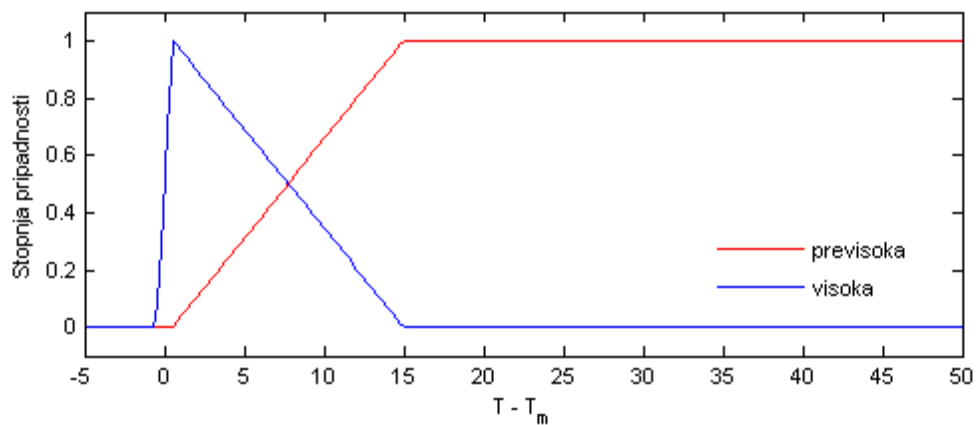
P_2 je spremenljivka razlike med temperaturo stene in želeno temperaturo $T_s - T$ in prav tako vsebuje štiri mehke množice. Množici *pozitivna* in *negativna* vklopita oziroma izklopita grelce, delovanje množic *pozitivna_okolica* in *negativna_okolica* pa je odvisno od območja spremenljivke P_1 .

Za določitev vrednosti množic v obeh pripadnostnih funkcijah je bila potrebna serija testiranja. Najprej je bilo potrebno določiti interakcijo med mehkim množicama *previsoka* in *visoka* v spremenljivki P_1 . Prehod iz maksimalnega segrevanja, torej, ko so vklopljeni vsi grelci na optimalno število grelcev, ki skrbijo da temperatura stene ne naraste preko dovoljenih mej. Prehod se začne z vsemi petimi grelci vklopljenimi in nato število vključenih grelcev postopoma zmanjšuje. Slednje povzroči manjšo inercijo sistema, kot je v že obstoječem algoritmu. Ugotoviti bilo je potrebno kdaj se ta prehod začne in koliko časa traja. V ta namen so bila opravljena testiranja različnih postavitev omenjenih mehkih množic. V nadaljevanju so predstavljeni trije rezultati vključno z rezultatom uporabljenim v končnem algoritmu.

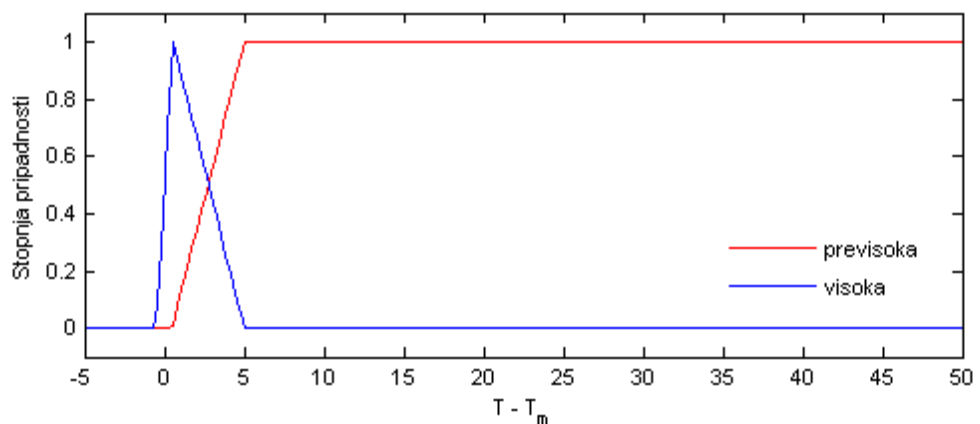


Slika 4.6 Interakcija mehkih množic *previsoka* in *visoka* na širokem prehodu.

Na sliki 4.6 sta prikazani mehki množici *previsoka* in *visoka* v širokem prehodu. Trajanje prehoda je v tem primeru najdaljše in sicer prehajanje iz segrevanja v vzdrževanje se začne, ko je razlika med želeno temperaturo in temperaturo medija $T - T_m$ enaka 25°C in traja do točke, ko je $T - T_m = 0.5^\circ\text{C}$. Celotno območje prehoda je torej 24.5°C . Slika 4.7 prikazuje srednji prehod med *previsoka* in *visoka*. Začne se, ko je $T - T_m = 15^\circ\text{C}$ in traja do razlike $T - T_m$ pri 0.5°C . Območje prehoda je 14.5°C . Na sliki 4.8 je prikazano ozki prehod, ki se začne pri razliki $T - T_m = 5^\circ\text{C}$ in se prav tako, kot ostali dve območji, konča pri $T - T_m = 0.5^\circ\text{C}$. Območje prehajanja je v tem primeru 4.5°C .

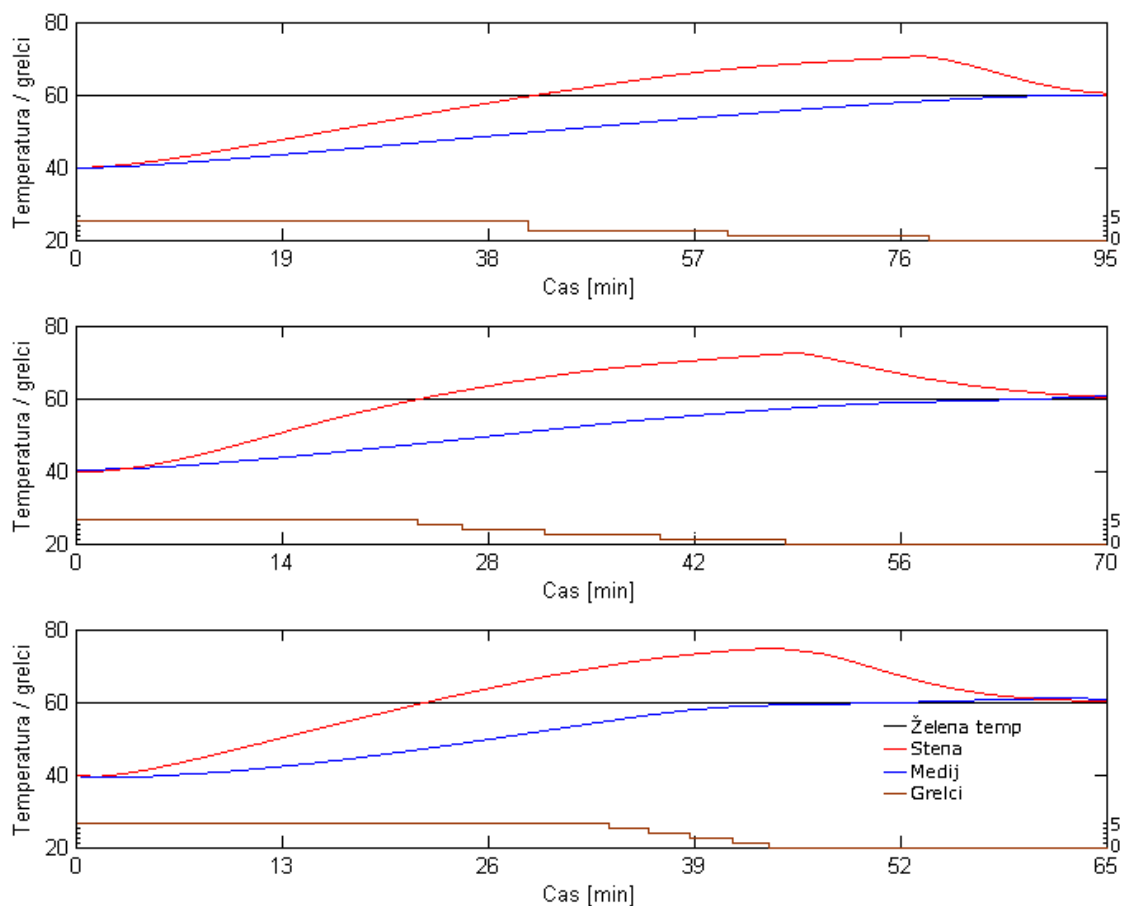


Slika 4.7 Interakcija mehkih množic *previsoka* in *visoka* na srednjem prehodu.



Slika 4.8 Interakcija mehkih množic *previsoka* in *visoka* na ozkem območju.

Grafi na sliki 4.9 prikazujejo rezultate testiranja za vsako od zgoraj opisanih možnosti. Začetna temperatura segrevanja medija je v vseh primerih v okviru 40°C , želena temperatura pa je 60°C . Tako so vsi rezultati v istem temperaturnem območju. Rezultate testiranja lahko vidimo tudi v tabeli 4.2.



Slika 4.9 Primerjava segrevanja ob različnih postavitvah mehkih množic *previsoka* in *visoka* spremenljivke P_1 .

Pri širokem prehodu segrevanje na želeno temperaturo traja, kot je pričakovati, najdlje in sicer 85 minut. Daljše trajanje segrevanja je posledica le štirih vklopljenih grelcev na samem začetku segrevanja. Maksimalno odstopanje temperature stene od zelene temperature je v tem primeru 11°C .

Pri srednjem prehodu je že opazna razlika v primerjavi s širokim prehodom. Ker ob začetku segrevanja deluje vseh pet grelcev, je želena temperatura dosežena v 65 minutah. Odstopanje temperature stene od zelene temperature pa je v tem primeru 13°C in je

posledica postopnega zmanjševanja števila vključenih grelcev in je posledica postopnega zmanjševanja števila delujočih grelcev.

Primer ozkega prehoda pripelje temperaturo medija do zelene temperature v 50 minutah, vendar je odstopanje temperature stene od temperature medija 16°C , kar povzroči dvig temperature medija čez 61°C in tako prestopi dovoljeni prag napake. Čeprav v zadnjem primeru temperatura medija doseže zeleno temperaturo najhitreje, je bolj optimalna rešitev srednjega območja prehoda, saj se v tem primeru temperatura medija zadrži v območju dovoljene napake.

Tabela 4.2 Primerjava območij prehoda mehkih množic *previsoka* in *visoka* mehke spremenljivke P_1 .

Območje	Trajanje [min]	Dovoljeno območje napake
Široko	85	da
Srednje	65	da
Ozko	50	ne

Ker pri segrevanju ni uporabljen histerezni algoritem, lahko pride, zaradi nihlajev v temperaturi, do nezaželenega večkratnega preklapljanja. To se zgodi pri spremembi števila vključenih grelcev. Problem je rešen s funkcijo eksponentnega glajenja. Ta funkcija pretekle vhodne podatke o temperaturi uporabi za napoved naslednje vrednosti vhoda [4] in tako izniči morebitna nihanja vhoda in s tem odpravi problem nezaželenih preklpov. Glajenje deluje po formuli (4.14).

$$s_0 = x_0, s_{t+1} = \alpha x_t + (1 - \alpha)s_t, t \geq 0, \alpha = 0.2 \quad (4.14)$$

Pri čemer je s_1 začetna temperatura, s_{t+1} pa zglajena temperatura v času $t + 1$, ki je sestavljena iz zglajene vrednosti in temperature v času t . Temperaturo stene ponazarja spremenljivka x . Glajenje vsebuje še faktor glajenja α , ki pripada pogoju $0 \leq \alpha \leq 1$.

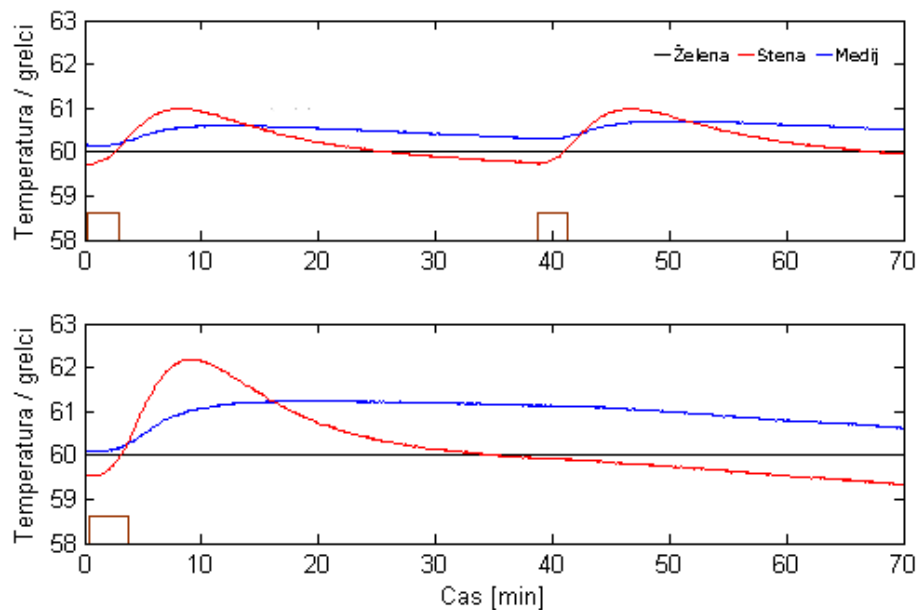
4.3.2 Spremenljivka P_2

Ko razlika $T - T_m$ preide v območje mehke množice *sprejemljivo*, se prične vzdrževanje temperature. Temperatura medija T_m vedno limitira proti temperaturi stene T_s , zato lahko slednjo uporabimo za ohranjanje zelene temperature vzdrževanja T . Mehke množice, ki skrbijo za vzdrževanje so del spremenljivke P_2 . Temperatura medija T_m mora pri vzdrževanju ostati v območju minimalne napake in sicer ne sme presegati T za več kot 1°C . Grelce je, zaradi inercije, potrebno izklopiti preden temperatura stene T_s doseže prag

napake. Mehka množica, ki predstavlja območje vzdrževanja temperature je *negativna_okolica*, množici *negativna* in *pozitivna_okolica* pa sta meji vzdrževanja. Ko razlika $P_2 = T_s - T$ doseže mehko množico *negativna*, se grelci vklopijo in sprožijo segrevanje. Ko s segrevanjem razlika $P_2 = T_s - T$ doseže mehko množico *pozitivna_okolica* se grelci izklopijo. Temu sledi krajše segrevanje medija zaradi temperaturne inercije.

Tabela 4.3 Mehke množice spremenljivke P_2 .

	Mehka množica	Definicija množice
	<i>negativna</i>	-10, -10, -0.5, -0.3
Primer 1	<i>negativna_okolica</i>	-0.5, -0.3, 0, 0.2
	<i>pozitivna_okolica</i>	0, 0.2, 0.6, 1
	<i>negativna</i>	-10, -10, -1, -0.5
Primer 2	<i>negativna_okolica</i>	-1, -0.5, 0, 1
	<i>pozitivna_okolica</i>	0, 1, 1.5, 2



Slika 4.10 Primer 1 iz tabele 4.3 je prikazan zgoraj, Primer 2 pa spodaj.

Grafa na sliki 4.10 prikazujeta pravilno upoštevanje inercije sistema (zgoraj) in povečanje napake vzdrževanja zaradi prepoznega izklopa grelcev (spodaj). Meje mehkih množic

negativna, *negativna_okolica* in *pozitivna_okolica* za oba primera so podane v tabeli 4.3. Vklon grelcev se v prvem primeru zgodi, ko je razlika $P_2 = T_s - T$ v območju $(-0.5, -0.3)$, izklop pa v območju $(0, 0.2)$. V drugem primeru je območje vklopa $(-1, 0.5)$, območje izklopa pa $(0, 1)$. V obeh primerih ob vklopu delujejo trije grelci. Tako vklop kot izklop grelcev je v primeru 1 bližje želeni temperaturi T . Zato se cikel vzdrževanja ponovi približno vsakih 40 minut, vendar je temperatura medija tudi ob izteku časovnega intervala 70 minut v dovoljenem območju napake ($T+1^\circ\text{C}$). V primeru 2 se cikel vzdrževanja izvede samo enkrat, vendar temperatura medija prekorači želeno temperaturo za 1.2°C , kar je za 0.2°C več od dovoljene napake.

Problem v primeru 2 je že omenjena inercija sistema, ki je prikazana tudi na grafih slike 4.10. Opaziti je, da se temperatura stene dviga kljub izklopljenim grelcem. Ker so bili v primeru 1 grelci pravočasno izklopljeni, temperatura medija ostane v dovoljenem območju medtem, ko v primeru 2 ta temperatura prekorači dovoljeno območje. Ker je v prvem primeru maksimalna temperatura stene, z upoštevanjem inercije, želeno temperaturo $T + 1^\circ\text{C}$, je temperatura medija ne more prekoračiti. V drugem primeru pa je, vključno z inercijo, temperatura stene celo višja od $T + 2^\circ\text{C}$, tako da je pričakovati dviganje temperature medija proti tej temperaturi v vsakem ciklu vzdrževanja. Tabela 4.4 prikazuje omenjene razlike.

Tabela 4.4 Rezultati vzdrževanja za oba primera.

	Max. T_s [$^\circ\text{C}$]	Max. T_m [$^\circ\text{C}$]	Čas cikla [min]	Št. grelcev
Primer 1	60.97	60.74	39	3
Primer 2	62.18	61.2	71	3

5 Rezultati

5.1 Primerjava rezultatov

V tem poglavju sledi primerjava dejanskega segrevanja in vzdrževanja z uporabo algoritma predstavljenega v poglavju 2.4 in algoritma mehke logike iz poglavja 4.2. Da so rezultati bolj primerljivi je bilo potrebno zagotoviti približno enako začetno temperaturo segrevanja v obeh primerih. Zelene temperature so prav tako iste. V poglavju 3 je bilo pokazano, da posamezne stopnje segrevanja in vzdrževanja vsebujejo rezultate primerljive z rezultati simulacije celotnega procesa. Tudi v tem delu velja ta ugotovitev in v ta namen primerjava rezultatov ni osredotočena na celoten proces ampak na posamezne stopnje segrevanja in vzdrževanja.

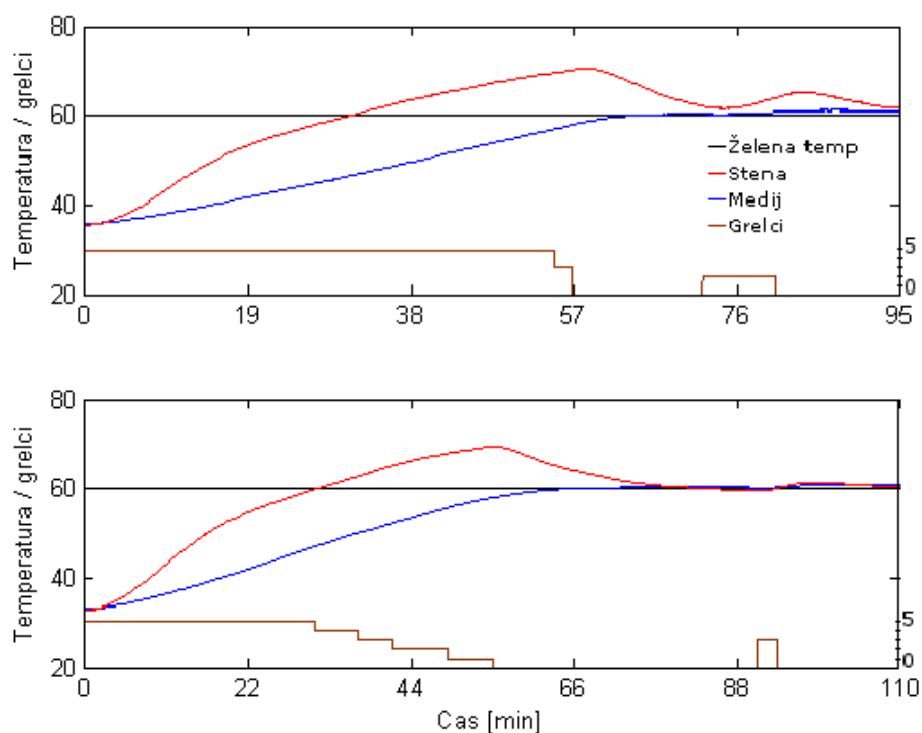
Na sliki 5.1 vidimo primerjavo segrevanja iz začetne temperature v okolici 37°C do zelene temperature 60°C . Na zgornjem grafu je prikazan potek segrevanja starega algoritma na spodnjem pa algoritma mehke logike.

Kot je razvidno tudi iz tabele 5.1, stara rešitev pripelje temperaturo medija od začetne do zelene temperature v približno 62 minutah, medtem ko nova rešitev za to potrebuje okoli 65 minut. Ta razlika v času je zanemarljiva glede na celoten potek procesa, vendar

Tabela 5.1 Primerjava segrevanja stare in nove rešitve.

Rešitev	Čas segrevanja [min]	Odstopanje T_s [°C]
Stara	62	13
Nova	65	10

pa se različen postopek segrevanja pokaže pri odstopanju temperature stene od zelene temperature in posledično pri nadaljnem vzdrževanju temperature. Nova rešitev ima manjše število vključenih grelcev, ko je temperatura medija v okolici zelene temperature, zato je tudi inercija sistema manjša.



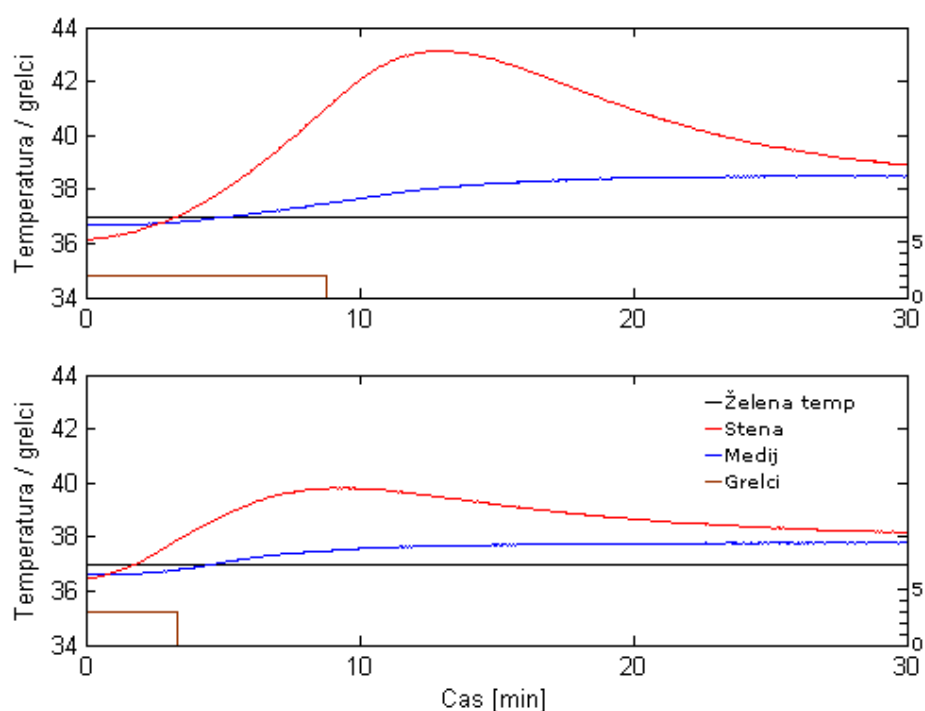
Slika 5.1 Primerjava starega (zgoraj) in novega (spodaj) algoritma - segrevanje na 60°C.

Na sliki 5.1 je poleg segrevanja prikazan tudi prvi cikel vzdrževanja pri katerem je opaziti, da se pri stari rešitvi na račun hitrejšega segrevanja (večino časa je prižganih vseh pet grelcev) temperatura medija pri vzdrževanju dvigne čez dovoljeni prag napake, ki je v tem primeru 61°C. Pri novi rešitvi pa temperatura medija ostane pod tem pragom.

Prav tako je nova rešitev v tem primeru energijsko varčnejša, saj se število vključenih grelcev postopoma manjša in tudi cikel vzdrževanja je krajši in se ponovi redkeje.

Tabela 5.2 Primerjava segrevanja v okolici zelene temperature.

Rešitev	Čas segrevanja [min]	Odstopanje T_{stene} [°C]	Max T_{medija} [°C]
Stara	5	2.9	38.3
Nova	4	6.2	37.9



Slika 5.2 Primerjava starega (zgoraj) in novega (spodaj) algoritma - segrevanje v okolici zelene temperature.

Slika 5.2 prikazuje segrevanje, ko je zelena temperatura v okolici začetne temperature. Oba primera prikazujeta potek temperatur na 30 minutnem intervalu. Na zgornjem grafu je prikazan potek segrevanja za staro rešitev, na spodnjem pa za novo. Čeprav pri stari rešitvi segrevanje poteka le z dvema grelcema, zaradi daljšega časovnega intervala dvigne inercija sistema temperaturo stene krepko preko zelene temperature in posledično preko dovoljene napake. V danem primeru temperatura stene naraste v okolico 43°C, medtem ko je zelena temperatura le 37°C. To povzroči tudi prekomerno rast temperature medija,

ki ob koncu 30 minutnega intervala naraste na 38.3°C , kar je za 0.3°C preko dovoljene napake. Pri novi rešitvi so za začetek segrevanja zadolženi trije grelci, vendar je interval segrevanja opazno krajši od intervala pri stari rešitvi. Zaradi tega temperatura stene naraste le do okolice 40°C in temperatura medija po 30 minutnem intervalu ostane v okviru dovoljene napake na 37.9°C . Stara rešitev doseže želeno temperaturo po 5 minutah, nova pa potrebuje 4 minute. Omeniti je potrebno da v tem primeru ne gre za vzdrževanje, ampak za segrevanje na želeno temperaturo. Pred tem je sistem v mirovanju. Rezultati so predstavljeni tudi v tabeli 5.2.

Sledi še primerjava vzdrževanja v procesu kuhanja. Vzdrževanje se prične, ko temperatura medija doseže želeno temperature. Na sliki 5.3 sta prikazana grafa vzdrževanja stare in nove rešitve za časovni interval 75 minut v tabeli 5.3 pa so povzeti rezultati vzdrževanja. Pri stari rešitvi se cikel vzdrževanja, to je vklop grelcev, ponovi trikrat. Gre za vklop dveh grelcev za približno 9 minutni interval in nato približno 10 minutno padanje temperature stene. Celoten cikel traja 19 minut. Temperatura stene se v danem primeru giblje od 97°C do 100°C , zato tudi temperatura medija ob vsakem ciklu naraste več kot v prejšnjem. Po končanem tretjem ciklu vzdrževanja, je temperatura medija že preko dovoljene napake in sicer na 96.4°C . V primeru da bi se vzdrževanje nadaljevalo, bi se napaka še povečevala.

Tabela 5.3 Primerjava vzdrževanja temperature stare in nove rešitve.

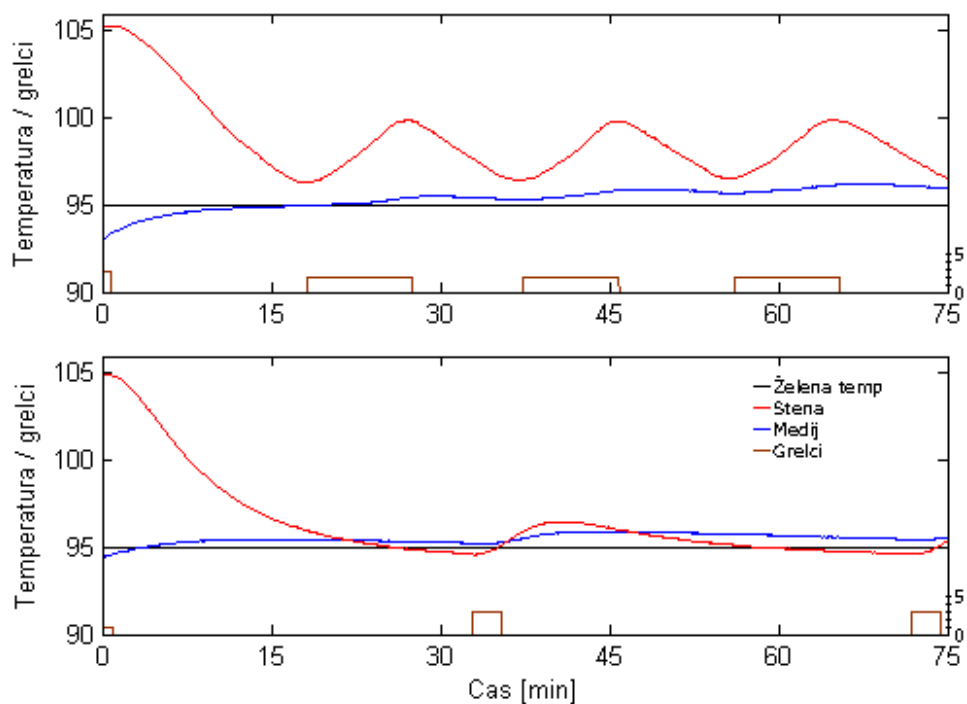
Rešitev	Čas cikla [min]	Odstopanje T_{stene} [$^{\circ}\text{C}$]	Max T_{medija} [$^{\circ}\text{C}$]
Stara	19	5	96.4
Nova	40	1.2	95.7

Cikel vzdrževanja se pri novi rešitvi ponovi vsakih 40 minut. Za 2.5 minutni interval se vklopijo trije grelci. Inercija sistema je ob izklopu grelcev večja kot pri stari rešitvi, a zaradi kratkega intervala segrevanja ne naraste preko omejitev. Temperatura medija tako ostane v območju dovoljene napake. Temperatura stene v vzdrževalnem ciklu pade pod temperaturo medija in s tem zagotavlja primerno temperaturo stene. Ob zaključku 75 minutnega opazovanja, je temperatura medija 95.7°C , kar je dopustno glede na maksimalno napako, ki je 1°C .

Iz podanih rezultatov je razvidno da ima pri prvi primerjavi segrevanja stara rešitev zanemarljivo prednost pri hitrosti doseganja želene temperature. Nova rešitev zagotavlja obstoj temperature medija v dovoljenih mejah, medtem ko stara rešitev maksimalno

dovoljeno napako prekorači. Pri segrevanju, ko je želena temperatura v okolici začetne, je boljša nova rešitev, ki prav tako upošteva določeno maksimalno napako, medtem ko je stara rešitev ne. Časovna razlika pri segrevanju je v tem primeru zanemarljiva. Tudi pri vzdrževanju je v prednosti nova rešitev. Pri stari rešitvi se temperatura medija postopoma odmika od želene temperature in naposled tudi prekorači dovoljeno napako, medtem ko nova rešitev temperaturo medija ohranja v okolici želene temperature.

Nova rešitev je boljša tudi iz vidika varčevanja z energijo. Pri stari rešitvi segrevanje temelji na vseh grelcih, pri novi, pa se število vključenih grelcev zmanjšuje, ko temperatura medija prihaja v okolico želene temperature. Tudi cikli vzdrževanja so pri novi rešitvi krajši kot pri stari in posledično, potrebujemo manj energije za isti oziroma celo boljši učinek.



Slika 5.3 Primerjava stare (zgoraj) in novega (spodaj) rešitve - vzdrževanje temperature.

Iz rezultatov je opaziti, da potek segrevanja precej podoben pri obeh algoritmih. Zato je potrebna še primerjava energijske porabe obeh algoritmov. V tabeli 5.4 je prikazano povprečje delujočih grelcev G_{avg} , njihova povprečna moč P_{avg} in delo W , ki je potrebno

pri segrevanju in vzdrževanju za obe rešitvi. Pri segrevanju se upošteva interval od začetka segrevanja do dosežene zelene temperature (tabela 5.1), pri vzdrževanju pa gre v obeh primerih za 75 minutni interval. Enačba za izračun dela:

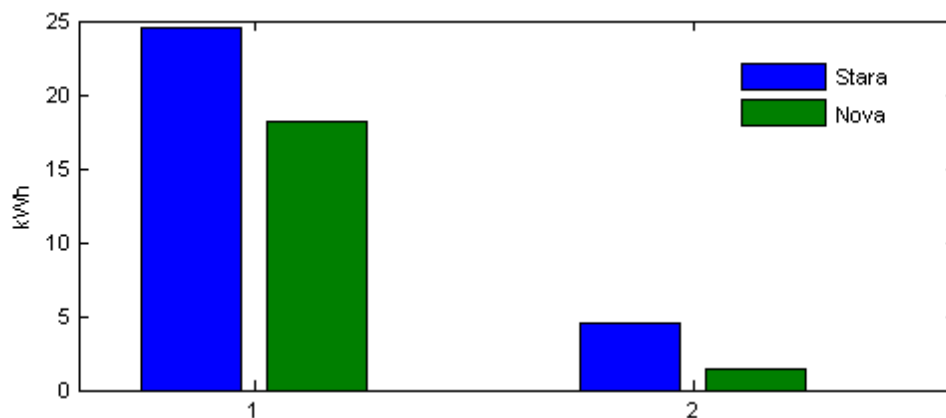
$$W = (5kW * G_{avg}) * t \quad (5.1)$$

Pri čemer je $5kW$ moč enega grelca, G_{avg} povprečje delujočih grelcev in t trajanje segrevanja ali vzdrževanja.

Tabela 5.4 Poraba grelcev pri segrevanju na 60°C in vzdrževanju na 60°C .

Proces	Rešitev	G_{avg}	P_{avg} [kW]	W [kWh]
Segrevanje	Stara	4.75	23.75	24.54
	Nova	3.36	16.8	18.19
Vzdrževanje	Stara	0.73	3.65	4.56
	Nova	0.22	1.1	1.37

Kot je razvidno tudi iz slike 5.4, nova rešitev prihrani energijo tako pri segrevanju kot tudi pri vzdrževanju. Pri segrevanju je prihranek dela 6.35 kWh, pri vzdrževanju pa 3.19 kWh. Na račun boljše energijske izkoriščenosti, se lahko novemu algoritmu spregleda malenkost počasnejše doseganje zelene temperature, kot je prikazano v tabeli 5.1.



Slika 5.4 Primerjava opravljenega dela W stare in nove rešitve pri segrevanju in vzdrževanju

6 Zaključek

6.1 Sklepne ugotovitve

V diplomskem delu sem poskušal razviti splošni algoritem segrevanja in vzdrževanja temperature vode ter pivine v procesu varjenja piva. Namen algoritma je izboljšanje natančnosti segrevanja in optimizacija energijske porabe pri samem procesu. Na začetku je predstavljen razvojni sistem, ki služi tudi za dejansko kuhanje piva, ključne komponente sistema in tudi ostala orodja, kot pomemben člen razvojnega okolja. Začetna poglavja vsebujejo tudi dejanski proces pridelave piva na omenjenem sistemu in opis stare rešitve segrevanja na podlagi katere je bilo izvršeno zajemanje osnovnih podatkov o sistemu. Gre za časovne in temperaturne karakteristike segrevanja in vzdrževanja temperature vode in pivine.

Analiza zajetih podatkov je podprta z grafi in tabelami, ki prikazujejo potek temperature stene, temperature medija in števila vključenih grelcev v odvisnosti od časa. Vključena so posamezna segrevanja na določeno temperaturo, vzdrževanje ob določeni temperaturi in simulacija ter dejanski proces kuhanja. Prav tako je izvršena analiza segrevanja z različnim številom vključenih grelcev v enakem časovnem intervalu. Po-

membna je bila ugotovitev, da so poteki posameznih segrevanj med seboj podobni in odražajo lastnosti dejanskega procesa kuhanja.

Sledi opis mehke logike in razvoj mehkega algoritma. Predstavljene so pripadnostne funkcije, pripadajoča mehka pravila in metoda defuzifikacije. Mehka logika je v tem primeru izbrana zaradi razširitve in nadgradnje ostrih množic in prikazuje nekonvencionalno rešitev opisane problematike.

Rezultati starega in novega algoritma so zbrani v svojem poglavju. Da je nova implementacija algoritma boljša od stare pričajo grafi in tabele, na katerih je razvidno, da stari algoritem ne more zagotoviti obstoja temperature medija v dovoljenem okviru napake, medtem ko nova rešitev to omogoča. Mehki algoritem predstavlja tudi izboljšanje energijske porabe pri procesu kuhanja, saj upošteva optimalno število vključenih grelcev in krajše časovne intervale segrevanja.

Dodati je še potrebno, da se mehka logika izkaže za zelo uporabno v primeru segrevanja, ko se dinamično spreminja število vključenih grelcev. Pri tem pa je potrebno zagotoviti glajenje vhodnih podatkov, ki prepreči nezaželen vklop in izklop grelcev zaradi manjšega nihanja temperature. Pri vzdrževanju mehka logika ne igra pomembne vloge, saj bi lahko ta del algoritma nadomestili tudi z ustreznimi ostrimi mejami.

6.2 Predlogi za nadaljnje delo

Nadaljnje delo je mogoče tako na algoritmu, kot tudi na sistemu samem. V primeru izgradnje sistema z večjim volumnom kotla, bo potrebno analizirati rezultate segrevanja na takem sistemu in ugotoviti ustreznost algoritma. Večji kotli imajo v večini primerov drugačne tehnike segrevanja in mešanja medija, ki bi potencialno lahko vplivali na delovanje algoritma.

Še en mogoč nadaljnji korak pri razvoju algoritma je implementacija logike za hlajenje. Algoritem v tem diplomskem delu se ukvarja le s segrevanjem in vzdrževanjem temperature. Potrebno bi bilo še nekako zagotoviti delovanje v primeru, ko je potrebno temperaturo znižati s pomočjo hladilnega medija. Algoritem bi moral prepoznati ali gre za segrevanje, vzdrževanje ali hlajenje in izvršiti ustrezno akcijo.

Izboljšave so možne tudi na samem sistemu neodvisno od algoritma. Prva možna izboljšava je pozicija temperaturnega senzorja stene. Ker se ta senzor nahaja med spodnjima dvema grelcema, je včasih temperatura stene zavaajajoč podatek, saj se nanaša

la na spodnji del kotla. Rešitev lahko vključuje tudi dodatne senzorje in primerjavo temperatur med njimi.

Ena izmed očitnih izboljšav sistema je način segrevanja. Trenutno se segrevanje vrši preko električnih grelcev, ki pa sistemu povzročijo opazno inercijo, tudi, ko so že izklopljeni. Poraba električne energije na takih grelcih nikakor ni zanemarljiva. Možna rešitev je gretje s pomočjo olja ali pare, ki sistemu dodajo večjo odzivnost in manjšo inercijo.

LITERATURA

- [1] J. P. Arnold, Origin and History of Beer and Brewing: From Prehistoric Times to the Beginning of Brewing Science and Technology, Chicago: Alumni Association of the Wahl-Henius Institute of Fermentology., 1911.
- [2] J. R. A. Pollock, Brewing Science, Vol. I, II, III, Academic press, 1979.
- [3] I. Lebar Bajec, Fuzzy model for a computer simulation of bird flocking, PhD dissertation, University of Ljubljana, Faculty of Computer and Information Science (2005).
- [4] R. G. Brown, Smoothing Forecasting and Prediction of Discrete Time Series, Englewood Cliffs, NJ: Prentice-Hall, 1963.