

UVOD U MODELSKO PREDIKTIVNO UPRAVLJANJE

Gavranić, Ivana

Undergraduate thesis / Završni rad

2021

Degree Grantor / Ustanova koja je dodijelila akademski / stručni stupanj: **Josip Juraj Strossmayer University of Osijek, Faculty of Electrical Engineering, Computer Science and Information Technology Osijek / Sveučilište Josipa Jurja Strossmayera u Osijeku, Fakultet elektrotehnike, računarstva i informacijskih tehnologija Osijek**

Permanent link / Trajna poveznica: <https://um.nsk.hr/um:nbn:hr:200:831192>

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

Download date / Datum preuzimanja: **2025-02-23**

Repository / Repozitorij:

[Faculty of Electrical Engineering, Computer Science and Information Technology Osijek](#)



**SVEUČILIŠTE JOSIPA JURJA STROSSMAYERA U OSIJEKU
FAKULTET ELEKTROTEHNIKE, RAČUNARSTVA I
INFORMACIJSKIH TEHNOLOGIJA**

Sveučilišni studij

UVOD U MODELSKO PREDIKTIVNO UPRAVLJANJE

Završni rad

Ivana Gavranić

Osijek, 2021.

SADRŽAJ

1. UVOD	1
1.1. Zadatak završnog rada	1
2. MODELSKO PREDIKTIVNO UPRAVLJANJE.....	2
2.1. Opća strategija modelskog prediktivnog upravljanja	2
2.2. Komponente modelskog prediktivnog upravljanja	5
2.2.1. Matematički model procesa	5
2.2.2. Kriterijska funkcija	6
2.2.3. Ograničenja.....	6
2.2.4. Optimizacijski problem	7
2.3. Prednosti i nedostaci	7
2.4. Tipovi modelskog prediktivnog upravljanja	9
3. PRIMJER PROJEKTIRANJA I SIMULACIJSKI REZULTATI	11
3.1. Matematički model vozila.....	11
3.2. Parametri procesa i ograničenja.....	13
3.3. Koraci za dobivanje modela pomoću Simulinka.....	14
3.4. Rezultati	16
4. ZAKLJUČAK.....	19
LITERATURA	20
SAŽETAK.....	21
ABSTRACT	22

1. UVOD

Modelsko prediktivno upravljanje (engl. *Model predictive control*, ili skraćeno MPC) predstavlja popularno upravljanje koje inženjeri koriste u procesnoj industriji od 1980-ih. Njegova se upotreba proširila na mnoga područja, jedna od njih je automobilska industrija[2]. MPC koristi model za predviđanje budućeg ponašanja sustava kojim upravlja te kako bi pronašao najbolji mogući put za rješavanje problema. Može upravljati sustavima s više ulaza i izlaza, tj. MIMO sustavima (engl. *Multiple-input multiple-output*). Jedna od većih prednosti MPC-a je ta što sadrži ograničenja jer prilikom kršenja istih dolazi do neželjenih posljedica.

Kako bi uspješno upravljali sustavom pomoću MPC-a potrebno je pažljivo odabrati njegove parametre.

U drugom poglavlju dana je teorijska pozadina MPC-a. Dana je opća strategija modelskog prediktivnog upravljanja i zatim su opisane njegove glavne komponente. Predstavljene su bitne prednosti MPC-a u odnosu na ostale metode upravljanja te neki od najvažnijih tipova MPC-a. U trećem poglavlju opisan je matematički model vozila te su prikazani rezultati simulacije upravljanja pomoću MPC-a. Također je objašnjeno kako navesti parametre u aplikaciji MPC Designer. Četvrto poglavlje predstavlja zaključak u kojem se rezimiraju postignuti rezultati.

1.1. Zadatak završnog rada

Kao određeno proširenje gradiva kolegija Osnove automatskog upravljanja, potrebno je sačiniti uvod u modelsko prediktivno upravljanje. Pri tome je potrebno dati osnovna načela ovakvog načina upravljanja i matematičku podlogu za projektiranje regulatora te to ilustrirati kroz prikladan primjer. Primjer realizirati simulacijom sustava upravljanja koristeći programski paket Matlab/Simulink.

2. MODELSKO PREDIKTIVNO UPRAVLJANJE

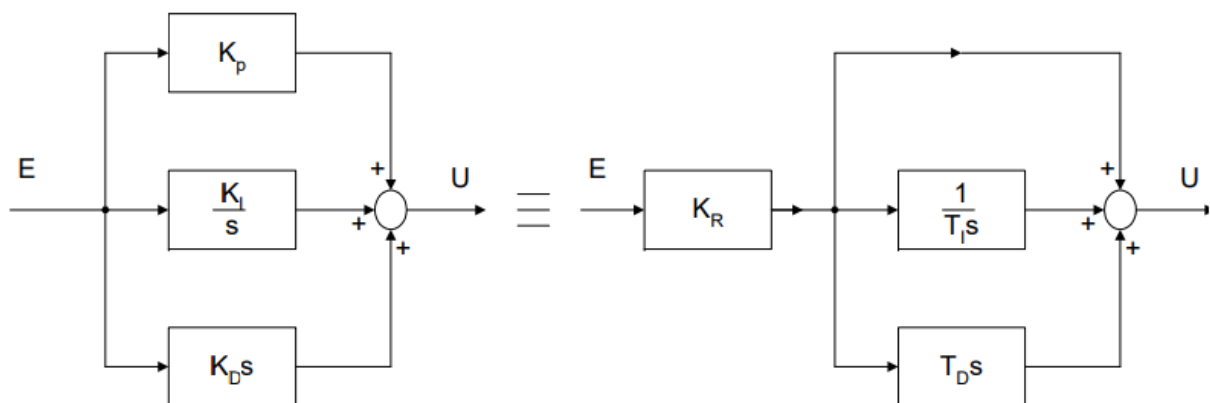
U ovom poglavlju iznesena je opća strategija modelskog prediktivnog upravljanja te je dan kratak pregled najvažnijih svojstava modelskog prediktivnog upravljanja. Zatim su pobliže opisane glavne komponente MPC-a, njegove prednosti i nedostaci. Na kraju poglavlja dan je kratak pregled ostalih tipova modelskog prediktivnog upravljanja.

2.1. Opća strategija modelskog prediktivnog upravljanja

Svim algoritmima modelskog prediktivnog upravljanja zajednički je osnovni koncept[2]:

- 1) korištenje matematičkog modela procesa u cilju predikcije regulirane veličine procesa tijekom određenog budućeg vremena (predikcijskog horizonta),
- 2) proračun upravljačke sekvence, koja se dobiva minimizacijom kriterijske funkcije,
- 3) strategija pomičnog horizonta koja podrazumijeva pomicanje predikcijskog horizonta nakon svakog vremena uzorkovanja za jedan korak u budućnost, te ponavljanje izračuna upravljačkog signala.

Jedan dio gradiva gradiva iz kolegija Osnove automatskog upravljanja vezan je za PID regulator[6]. U nastavku će se predstaviti standardni PID regulator.



Slika 2.1. PID regulator.

$$K_R = K_P$$

$$T_I = \frac{K_P}{K_I}$$

$$T_D = \frac{K_D}{K_P}$$

U praksi se danas upotrebljavaju regulatori koji se zasnivaju na P, I i D djelovanju. Na slici 2.1 predstavljen je PID regulator koji sadrži koeficijent pojačanja K_R , integralnu vremensku konstantu T_I te derivacijsku vremensku konstantu T_D .

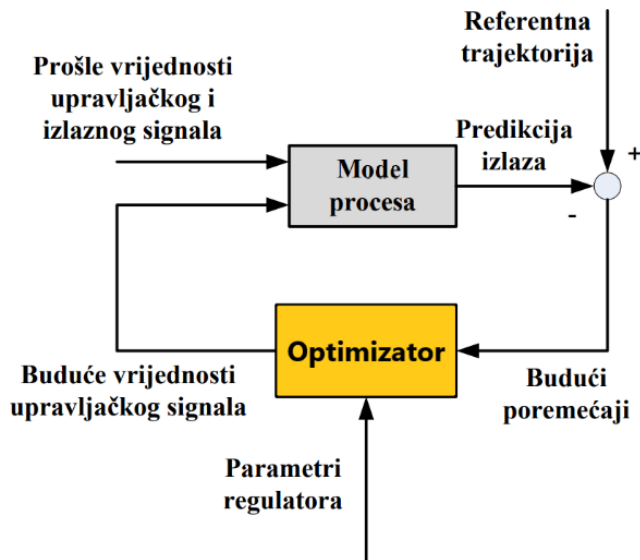
PID regulator se može prilagoditi vladanju procesa izborom podesivih parametara regulatora tako da se postigne najpovoljnije regulacijsko vladanje sustava. Korištenjem PID regulatora upravljačke petlje se izvode neovisno jedna o drugoj. Projektiranjem većih sustava povećava se broj parametara koji zahtjevaju usklađivanje. Kako bi se savladao taj problem kositi se MPC koji može upravljati sustavima s više ulaza i izlaza uzimajući u obzir varijable cijelog sustava[4].

Metoda modelskog prediktivnog upravljanja se odvija tako da se u svakom diskretnom vremenskom koraku k izračunava predikcija izlaza procesa za definirani predikcijski horizont N , a sve na osnovi: matematičkog modela procesa, vektora prošlih upravljačkih (ulaznih) vrijednosti \vec{u} , vektora prošlih vrijednosti regulirane veličine (izlaza) procesa \vec{y} i predikcijskog vektora upravljačke veličine \vec{u}^* kojeg tek treba odrediti. Predikcijski vektor upravljačke veličine proračunava se minimizacijom određene kriterijske funkcije koja ima za cilj dovesti buduće vrijednosti izlaza procesa \vec{y} , što je moguće bliže referentnoj trajektoriji \vec{r} . Nakon što se izračuna predikcijski vektor upravljačke veličine:

$$\vec{u}^*(k) = [u^*(k) \ u^*(k+1) \ \dots \ u^*(k+N-1)], \quad (2.1.)$$

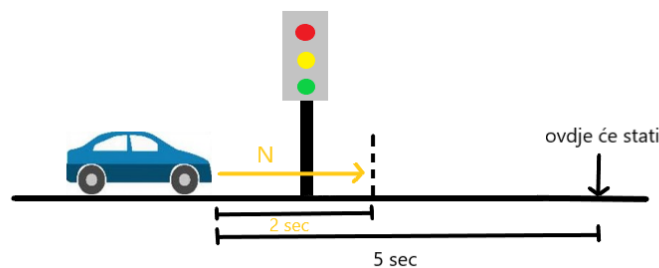
njegov prvi element $u^*(k)$ šalje se na izlaz regulatora, dok se svi ostali elementi zanemaruju i odbacuju. Postupak proračuna se ponavlja i u slijedećem vremenskom koraku, što čini princip tzv. pomičnog horizonta (engl. Receding Horizon).

Struktura poopćenog modelskog prediktivnog regulatora prikazana je na slici 2.2. Kao što se vidi sa slike, model procesa, pomoću prošlih vrijednosti upravljačkog i izlaznog signala, te budućih vrijednosti upravljačkog signala proračunava buduće vrijednosti izlazne veličine procesa.



Slika 2.2. Struktura poopćenog modelskog prediktivnog regulatora.

Odabiru veličine predikcijskog horizonta N treba pažljivo pristupiti, jer i u slučaju da je N premalen ili prevelik može doći do neželjenog ponašanja sustava[3]. Primjerice, automobil za koji se zna da ukoliko vozi 50 kilometara na sat da će trebati 5 sekundi da se zaustavi nakon pritiskanja kočnice. U slučaju da je automobil upravljani MPC-om i da je predikcijski horizont premalen, $N = 2$, automobil se neće zaustaviti do semafora (slika 2.3).



Slika 2.3. Odabir predikcijskog horizonta.

Iz priloženog metodologija MPC algoritama se može podijeliti u sljedeće komponente[7]:

- 1) matematički model procesa,

- 2) kriterijska funkcija,
- 3) ograničenja,
- 4) optimizacijski problem.

2.2. Komponente modelskog prediktivnog upravljanja

Komponente modelskog prediktivnog upravljanja su matematički model procesa, kriterijska funkcija, ograničenja te optimizacijski problem. U nastavku je svaka komponenta pobliže predstavljena.

2.2.1. Matematički model procesa

Najvažnija komponenta svakog MPC-a je matematički model procesa. Model procesa se koristi za predviđanje budućih izlaza i stanja procesa temeljem trenutnih stvarnih ili estimiranih vrijednosti. Model procesa se može odrediti teorijskom analizom (modeliranjem procesa), ali se češće određuje eksperimentalnom analizom (identifikacijom procesa). Važno je istaknuti da je točnost matematičkog modela sustava iznimno važna, jer u protivnom može doći do pogrešne predikcije budućih varijabli stanja te narušenih performansi upravljanja, kao i neželjenog vladanja sustava. Uobičajen zapis matematičkog modela procesa je matrični zapis u prostoru stanja[7]:

$$\frac{dx(t)}{dt} = \mathbf{f}(x(t), u(t)), \quad (2.2.)$$

$$y(t) = \mathbf{h}(x(t), u(t)), \quad (2.3.)$$

gdje je \mathbf{f} funkcija koja opisuje dinamiku sustava, a \mathbf{h} funkcija koja definira izlazne veličine $y(t)$ na temelju stanja procesa $x(t)$ i upravljačke (ulazne) veličine $u(t)$. Jednadžbe 2.2 i 2.3 vrijede i za linearne i nelinearne sustave. Posebno za linearne sustave model procesa u prostoru stanja ima oblik:

$$\frac{dx(t)}{dt} = \mathbf{A}x(t) + \mathbf{B}u(t), \quad (2.4.)$$

$$y(t) = \mathbf{C}x(t) + \mathbf{D}u(t), \quad (2.5.)$$

gdje su matrice **A**, **B**, **C**, **D** redom matrica dinamike sustava, ulazna matrica, izlazna matrica i matrica propusnosti. Jedna od prednosti ovog modela je prigodan za modeliranje procesa s više varijabli.

Osim modela stanja sustava, u praksi se koriste i model impulsnog odziva, prijenosna funkcija, prijelazna funkcija i drugi.

2.2.2. Kriterijska funkcija

Najčešće korištena kriterijska funkcija je kvadratna funkcija. Koristi se kao kvadrat razlike predviđenog izlaza i referentne vrijednosti. Oblik kriterijske funkcije značajno utječe na složenost MPC-a. Ukoliko kriterijska funkcija nije konveksna funkcija od u može se dogoditi da ne postoji optimalna upravljačka sekvenca ili da ju nije moguće odrediti u zadanom vremenskom periodu[7].

Opći oblik (kvadratne) kriterijske funkcije:

$$J = \vec{e}^T \mathbf{Q} \vec{e} + \vec{u}^T \mathbf{R} \vec{u}, \quad (2.6.)$$

$$\vec{e} = \vec{y} - \vec{r}, \quad (2.7.)$$

gdje je \vec{e} vektor pogreške, \vec{r} vektor referentnih vrijednosti, \vec{u} vektor ulaza, **Q**, **R** težinske matrice. Pomoću matrica **Q** i **R** definira se važnost određenih ciljeva. Veća komponenta na dijagonali matrice **Q** povlači bržu regulaciju k nuli odgovarajućeg izlaza. Pridjeljivanju većih vrijednosti određenom stanju treba oprezno pristupiti jer se samim time može smanjiti regularizacija drugog stanja, tj. izlaza. Matrica **R** je težinska matrica ulaza, tj. koristi se za pridavanje veće važnosti određenim ulazima.

2.2.3. Ograničenja

Svaki proces podliježe nekim ograničenjima. Primjerice, kod autonomne vožnje jedno od ograničenja u danoj situaciji može biti brzina automobila, održavanje sigurne udaljenosti od drugog automobila i dr. No, također jedno od ograničenja kod autonomne vožnje je maksimalan kut za koji se može okrenuti volan automobila. Prvo navedeno ograničenje predstavlja ograničenje na stanje sustava i takva su ograničenja najčešće definirana od strane korisnika. MPC regulator mora odlučiti o mogućnosti ispunjavanja tih ograničenja i ublažiti ih ako je potrebno. S druge strane, drugo navedeno ograničenje predstavlja ograničenje na upravljačke veličine i to ograničenje se mora ispuniti jer predstavlja fizičko ograničenje sustava. Ograničenja postavljena

na stanje sustava nazivaju se *blaga* ograničenja (engl. soft constraints), a ograničenja na upravljačke (ulazne) veličine nazivaju se *stroga* ograničenja (engl. hard constraints). Većina regulatora trenutno u upotrebi ne razmatra ograničenja eksplicitno. MPC, za razliku od ostalih regulatora, eksplicitno uključuje ograničenja u formulaciju svog optimizacijskog problema.

2.2.4. Optimizacijski problem

Rješenje optimizacijskog problema je vektor upravljačkih veličina. Vektor upravljačkih veličina osigurava minimum kriterijske funkcije na predikcijskom horizontu uz predikciju budućih varijabli stanja korištenjem matematičkog modela procesa uzimajući u obzir ograničenja varijabli stanja i upravljačke veličine.

Standardna formulacija optimizacijskog problema dana je sljedećim izrazom[7]:

$$\min_{\vec{u}(k)} J = \min_{\vec{u}(k)} (\vec{e}^T Q \vec{e} + \vec{u}^T R \vec{u}) \quad (2.8.)$$

$$\text{tako da } x(k+j+1) = f(x(k+j), u(k+j)), \text{ model procesa} \quad (2.9.)$$

$$x(k+j) \in \mathcal{X}, \text{ ograničenja na stanje procesa} \quad (2.10.)$$

$$u(k+j) \in \mathcal{U}, \text{ ograničenja ulaza} \quad (2.11.)$$

gdje je $j=0,1,\dots,N-1$, \mathcal{X} dopušteni skup varijabli stanja $x(k)$, \mathcal{U} dopušteni skup upravljačke (ulazne) veličine $u(k)$.

Premda je u svakom koraku izračunat vektor upravljačkih veličina na cijelom predikcijskom horizontu, vektor je dan jednadžbom 2.1, primjenjuje se samo optimalna upravljačka veličina u trenutnom periodu uzorkovanja $u^*(k)$ te se cijeli postupak ponavlja u narednom periodu uzorkovanja.

2.3. Prednosti i nedostaci

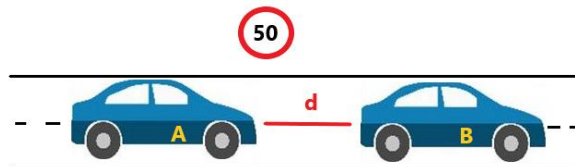
Važne prednosti MPC-a nad ostalim metodama upravljanja:

- 1) primijenjiv na procese značajno različitih vladanja: statičke, astatičke, minimalno i neminimalno fazne, s dobro ili loše prigušenim polovima itd.
- 2) upravlja izlazima simultano uzimajući u obzir interakcije između ulaza i izlaza;

- MPC se može jednostavno primijenjivati u procesima s više ulaza i više izlaza (engl. Multiple Input – Multiple Output systems).

3) poštivanje ograničenja postavljenih nad sustavom upravljanja;

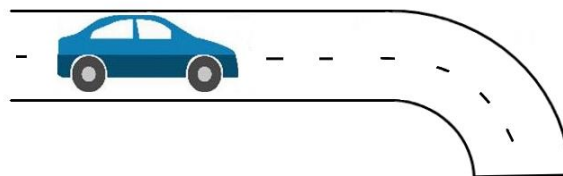
- ako je vozilo A autonomno vozilo (slika 2.4.) upravljano pomoću MPC-a, ono će pratiti željenu trajektoriju poštujući prije svega sigurnosna ograničenja prilikom vožnje, kao što je ograničenje na brzinu, održavanje sigurne udaljenosti od vozila B itd.



Slika 2.4. Sigurnosna ograničenja prilikom vožnje.

4) u potpunosti iskorištavanje saznanja o budućim trajektorijama referentne veličine;

- ako vozilo ima kameru koja pruža informacije o budućim trajektorijama regulator će unaprijed znati o nadolazećem zavoju pa može ranije usporiti i time poboljšati performansu upravljanja kretanjem vozila (slika 2.5.).



Slika 2.5. Vozilo i nadolazeći zavoj.

5) nije regulator već skup načela upravljanja pa se može nadograđivati i poboljšavati.

Iako ima brojne prednosti, MPC ima i svoje nedostatke. Neki od njih su:

- 1) potrebno je unaprijed poznavati model procesa (kakvoća upravljanja jako ovisi o kakvoći modela procesa),

- 2) kako bi model što bolje opisivao trenutno vladanje procesa, potrebno je provoditi on-line identifikaciju,
- 3) dobivanje upravljačkog algoritma je znatno složenije nego što je to slučaj kod klasičnih linearnih regulatora,
- 4) iz razloga 2) i 3) zahtjeva snažan, brz procesor s velikom memorijom jer MPC u svakom trenutku rješava *online* optimizacijski problem,
- 5) značajni matematički problemi u rješavanju optimizacijskog problema itd.

Međutim, zbog sve veće snage računala koji se koriste u upravljanjima procesa navedeni računarski zahtjevi MPC-a sve manje dolaze do izražaja. Kako bi se iskoristile sve prednosti MPC regulatora potrebno je identificirati što bolji matematički model procesa radi izbjegavanja matematičkog problema.

2.4. Tipovi modelskog prediktivnog upravljanja

Obzirom na linearnost modela procesa, na postavljena ograničenja i na definiranu kriterijsku funkciju MPC se grubo može podijeliti na linearan i nelinearan. U potpoglavlju 2.1. opisan je linearni MPC u kojemu su model procesa i ograničenja linearni, a kriterijska funkcija kvadratna. Vidi se da u tome slučaju optimizacijski problem je konveksan pa stoga postoji jedinstveno globalno rješenje. No postoje sustavi u kojemu model ili ograničenja nisu linearni, ili u kojemu definirana kriterijska funkcija nije kvadratna. U tim slučajevima MPC ima složeniji oblik.

Ukoliko su ograničenja linearna, kriterijska funkcija kvadratna, no proces nelinearan koji se može linearizirati tada se koristi adaptivni ili „Gain-Scheduled“ MPC. Postupak se sastoji od pronalaska više linearnih modela od kojih svaki interpolira nelinearan proces dobro u okolini njegove radne točke (engl. operating point). Također, u adaptivnom MPC-u broj stanja i ograničenja se ne mijenja za različite radne uvjete. Ukoliko se mijenjanju koristi se „Gain-Scheduled“ MPC. Kod „Gain-Scheduled“ MPC-a linearizacija u radnim točkama od interesa se obavlja *offline* te se u svakoj radnoj točki dizajnira linearan MPC regulator. Dizajnirani regulatori su međusobno neovisni i stoga mogu imati različit broj stanja i ograničenja. Također za „Gain-Scheduled“ MPC treba implementirati algoritam koji će provoditi zamjenu predefiniranih MPC regulatora obzirom na radne uvjete. Iako je postojanje većeg broja neovisnih MPC regulatora prednost ovoga tipa linearnog MPC-a, dodatna memorija potrebna za njihovo spremanje je veliki

nedostatak u odnosu na adaptivni MPC, ali je jednostavnija matematika pa su zahtjevi na procesor daleko manji[1].

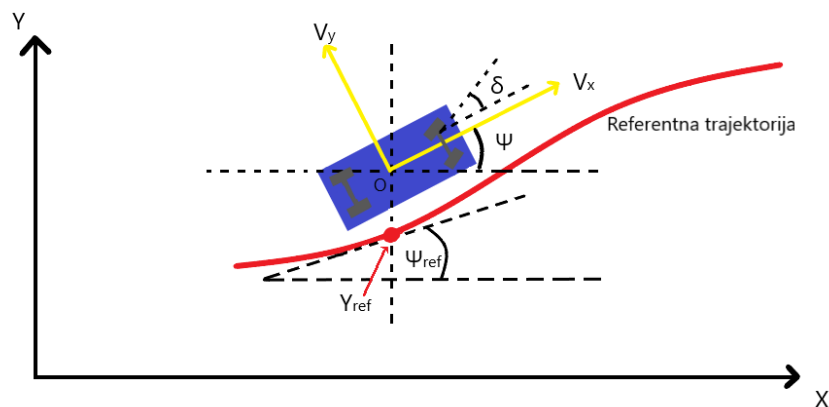
U slučaju da je sustav jako nelinearan, tj. ne može se linearizirati tada se koristi nelinearan MPC. U nelinearnom MPC-u optimizacijski problem je nekonveksan i može imati više lokalnih ekstrema pa traženje globalnog optimuma može biti izazovan zadatak. U takvim slučajevima koriste se razne iterativne metode numeričke matematike, a problem se najčešće svodi u formu neuronskih mreža.

3. PRIMJER PROJEKTIRANJA I SIMULACIJSKI REZULTATI

Praktični dio ovog rada odrađen je u programskom paketu MATLAB te Simulinku. Korištene su i aplikacije Driving Scenario Designer te MPC Designer koje su dio programa MATLAB. U radu je pokazano kako implementirati sustav automatski zakretanja vozila koristeći modelsko prediktivno upravljanje. U nastavku je dan opis korištenog modela vozila, parametara i ograničenja na sustav te rezultati simulacije. Primjer koji je korišten u ovom poglavlju nalazi se na poveznici [5].

3.1. Matematički model vozila

Kako se modelsko prediktivno upravljanje temelji na modelu procesa, najprije je potrebno modelirati vozilo. Skica gibanja vozila prikazana je na sljedećoj slici.



Slika 3.1. Skica gibanja vozila.

X i Y - položaj vozila u globalnim koordinatama,

V_x i V_y - longitudinalna i lateralna brzina vozila,

ψ - kut skretanja vozila,

δ - kut zakreta volana,

Y_{ref} - referentni lateralni položaj,

ψ_{ref} - referentni kut skretanja.

Kada se odabere struktura modela tada je potrebno odrediti parametre modela:

Tablica 3.1. Parameterni i vrijednosti modela

<i>Parametar</i>	<i>Vrijednost</i>
m	1575 kg
I_z	2875 kg m ³
I_f	1.2 m
I_r	1.6 m
C_f	19000 N/rad
C_r	33000 N/rad

m - masa vozila,

I_z - moment zakretanja inercije vozila,

I_f - udaljenost od težišta do prednjih kotača,

I_r - udaljenost od težišta do stražnjih kotač,

C_f - krutost prednjih guma u zavojima,

C_r - krutost stražnjih guma u zavojima.

U radu je korišten model vozila na dva kotača s dva stupnja slobode, lateralni položaj i kut skretanja vozila[2]. Model vozila na dva kotača poznatiji je pod nazivom *model bicikla* jer se u tome modelu prednji i stražnji par kotača reducira na jedan prednji te jedan stražnji kotač. U model vozila uključena je i lateralna dinamika vozila uz konstantnu longitudinalnu brzinu

$$V_x = 15.$$

Lateralna dinamika vozila može se reprezentirati koristeći linearni vremenski invarijantan sustav u kojem su:

- varijable stanja procesa: lateralna brzina V_y , kut skretanja vozila ψ , globalni Y položaj,
- ulazne varijable: kut zakretanja volana δ ,

- izlazne varijable: globalni Y položaj i kut skretanja vozila ψ .

Model lateralne dinamike vozila prikazan u prostoru stanja:

$$\frac{d}{dt} \begin{bmatrix} \dot{y} \\ \psi \\ \dot{\psi} \end{bmatrix} = \begin{bmatrix} -\frac{2C_f+2C_r}{mV_x} & 0 & -V_x - \frac{2C_fI_f-2C_rI_r}{mV_x} \\ 0 & 0 & 1 \\ -\frac{2I_fC_f-2I_rC_r}{I_zV_x} & 0 & -\frac{2I_f^2C_f+2I_r^2C_r}{I_zV_x} \end{bmatrix} \begin{bmatrix} \dot{y} \\ \psi \\ \dot{\psi} \end{bmatrix} + \begin{bmatrix} \frac{2C_f}{m} \\ 0 \\ \frac{2I_fC_f}{I_z} \end{bmatrix} \delta, \quad (3.1.)$$

$$\dot{Y} = V_y + V_x \psi. \quad (3.2.)$$

3.2. Parametri procesa i ograničenja

U odjeljku 2.2.2. vidjeli smo da je za definiranje kriterijske funkcije potrebno odrediti težinske matrice \mathbf{Q} , \mathbf{R} za ulazne i izlazne varijable. Za ovaj primjer odabrane su sljedeće vrijednosti:

$$\mathbf{Q} = \begin{bmatrix} 1 \\ 0.1 \end{bmatrix},$$

$$\mathbf{R} = [0].$$

Vidi se da je izlazna varijabla Y (1) ima veću težinu u odnosu na ψ (0.1) iz razloga što je praćenje pozicije vozila primarni zadatak ovog primjera.

Također, postavljena su sama ograničenja na ulazne i izlazne varijable sustava. Kao što je opisano ranije, na ulazne varijable postavljaju se stroga ograničenja, dok se na izlazne varijable postavljaju blaga ograničenja. U radu su ograničenja postavljena na sljedeći način:

- stroga ograničenja na kut zakretanja volana:

$$\delta \in \left[-\frac{\pi}{6}, \frac{\pi}{6}\right],$$

- blaga ograničenja na globalni položaj i kut skretanja vozila:

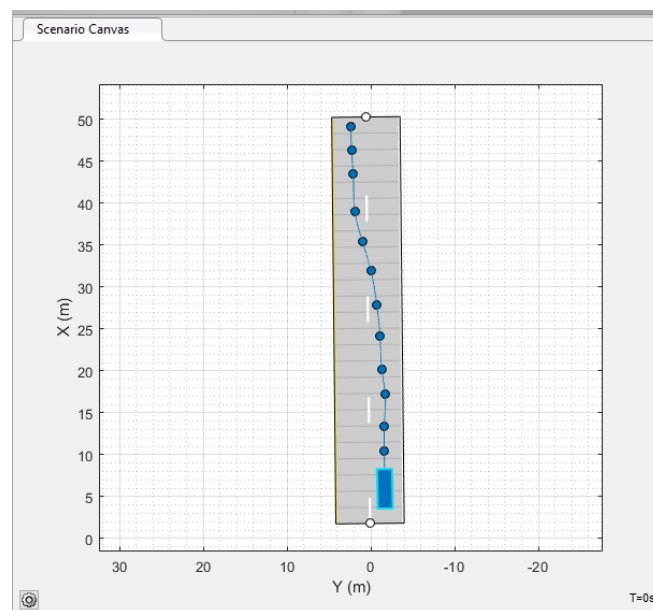
$$Y \in [-2, 6],$$

$$\psi \in [-0.2, 0.2].$$

Za predikcijski horizont N odabrana je vrijednost 10.

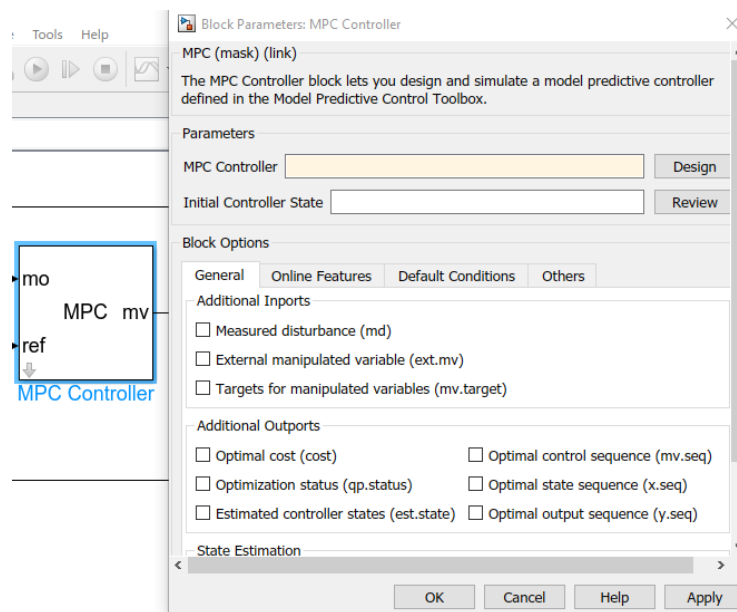
3.3. Koraci za dobivanje modela pomoću Simulinka

Na slici 2.3 prikazan je standardni dijagram upravljanja MPC-om koji se mora izgraditi[1]. Za izgradnju modela potrebno je imati proces. Proces ima jedan ulaz koji predstavlja kut zakreta volana te dva izlaza, a to su lateralni položaj Y i kut skretanja ψ . Sljedeći korak je dodati MPC regulator. Prvi ulaz u MPC regulator su izmjerni izlazi, a drugi ulaz je referenca. U ovom primjeru simulira se promjena trake vozila stoga je potrebno koristiti aplikaciju Driving Scenario Designer, kako bi se dobila prilagođena referentna putanja. Kada se otvori aplikacija prvo se postavlja cesta pa vozilo i zatim točke za generiranje manevra promjene trake, što prikazuje sljedeća slika.



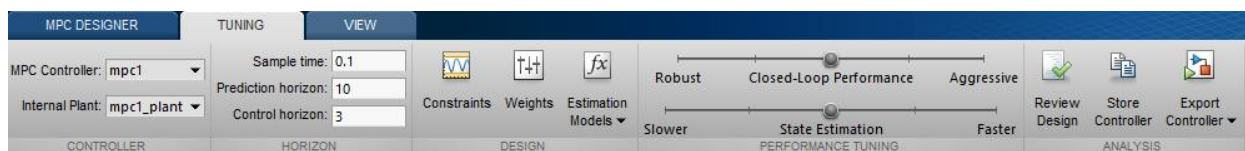
Slika 3.2. Prikaz scenarija manevra promjene trake.

Kao što je rečeno, longitudinalna brzina se postavlja na iznos 15 m/s. Kada se ovaj scenarij simulira korak po korak aplikacija pokazuje kako se mijenja kut skretanja. Nakon što se scenarij predstavi kao funkcija MATLAB stvori se blok koji prikazuje referentne vrijednosti lateralnog položaja i kuta skretanja. Blok se dodaje u model i povezuje sa regulatorom. MPC regulator ima i treći ulaz koji predstavlja smetnje no u ovom primjeru smetnje se zanemaruju te se može treći ulaz ukloniti. Sljedeće što se dodaje je Scope koji će dati odzive lateralnog položaja te referentnog lateralnog položaja, a isto vrijedi i za kut skretanja. Kako su sve komponente sustava spojene potrebno je projektirati MPC regulator. U tu svrhu otvara se MPC blok te se klikne na „Design“ koji pokreće MPC Designer.



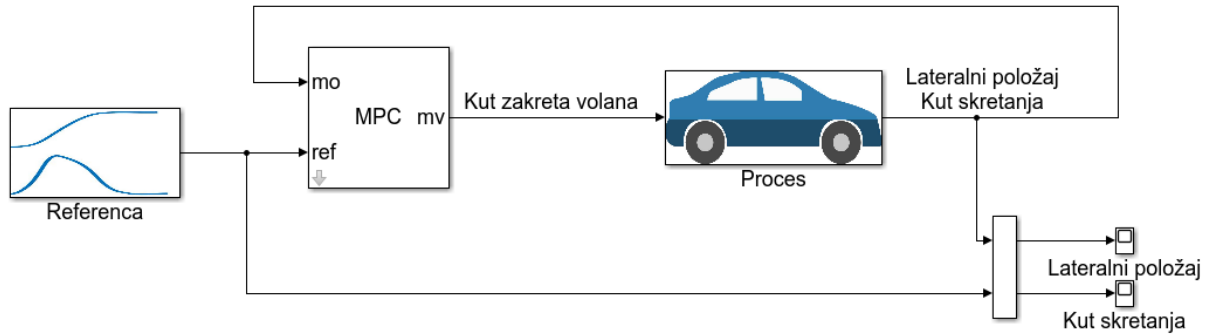
Slika 3.3. Korak ka aplikaciji MPC Designer.

Aplikacija MPC Designer omogućuje projektiranje MPC regulatora. Parametre MPC-a, kao što su vrijeme uzorkovanja, predikcijski horizont ograničenja i težine moguće je navesti u MPC Designer-u, podesiti regulator i zatim procijeniti performanse regulatora. MPC koristi interni model procesa za predviđanja, a optimizator za pronalaženje optimalne vrijednosti upravljačke veličine, tj. radnje upravljanja. Nakon što se postavi vrijeme uzorkovanja na 0,1 potrebno je kliknuti „definiraj i lineariziraj“. Kada se napravi taj korak aplikacija uvozi i linearizira proces iz modela Simulink i koristi ga kao interni model procesa. Kako bi se odredili ostali parametri MPC regulatora potrebno je prebaciti se na „Tuning“.



Slika 3.4. Prozor za mijenjanje parametara.

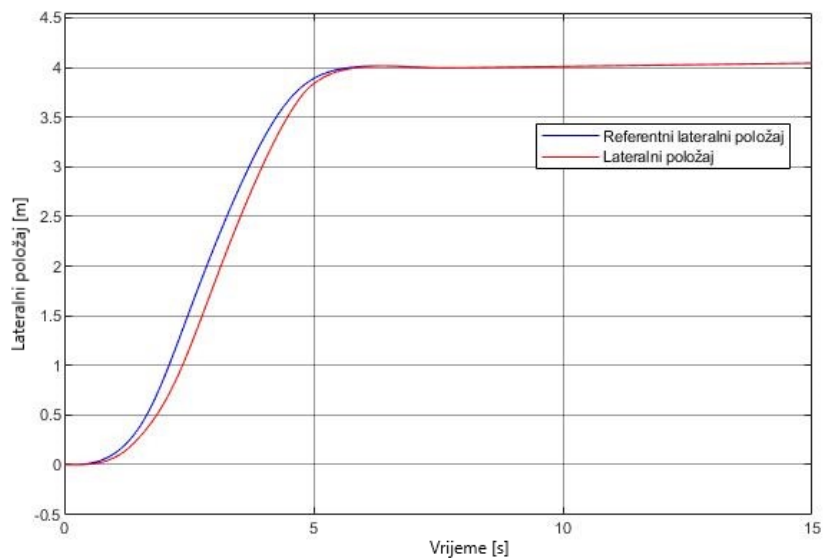
Važno je dobro podesiti parametre kako bi se moglo što bolje upravljati sustavom pomoću MPC-a. Kada je dobiven model u Simulinku može se pokrenuti simulacija. Prikaz sustava upravljanja dan je na sljedećoj slici.



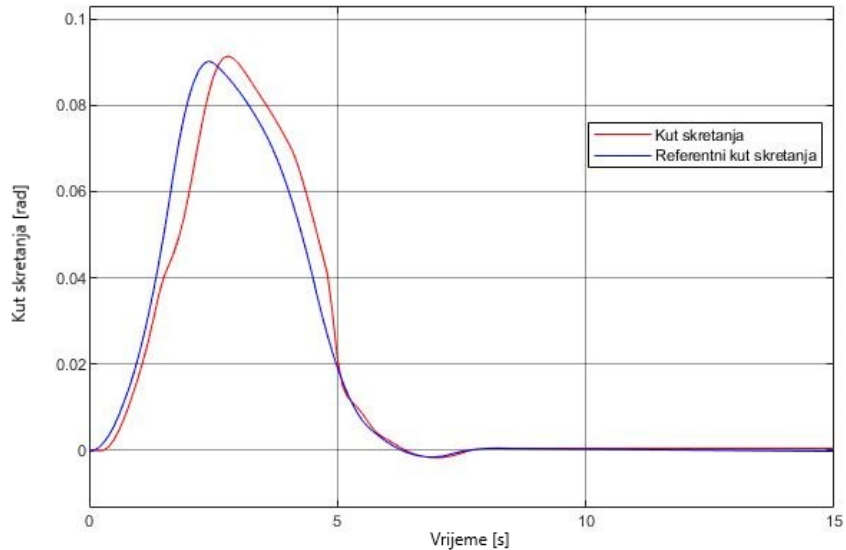
Slika 3.5. Sustav upravljanja.

3.4. Rezultati

Kada se pokrene simulacija dobiju se odzivi za oba izlaza, slike 3.6 i 3.7.



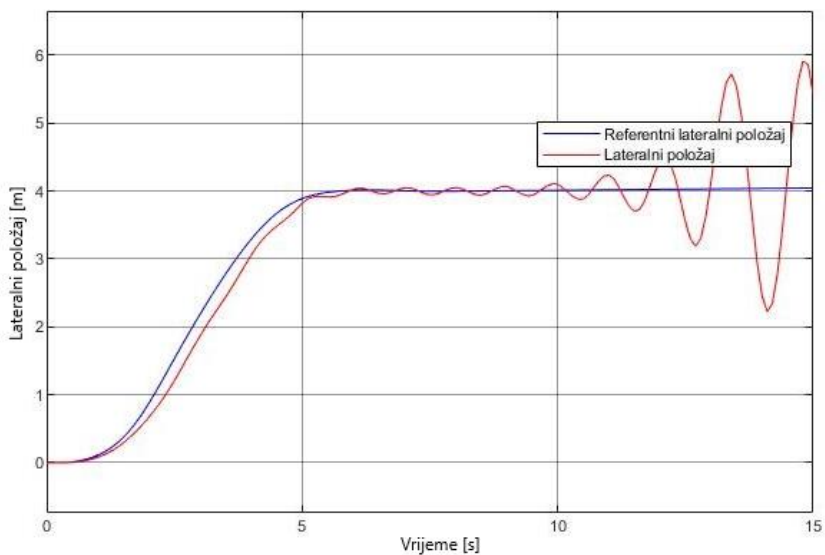
Slika 3.6. Prikaz odziva lateralnog položaja u slučaju kada je longitudinalna brzina 15 m/s.



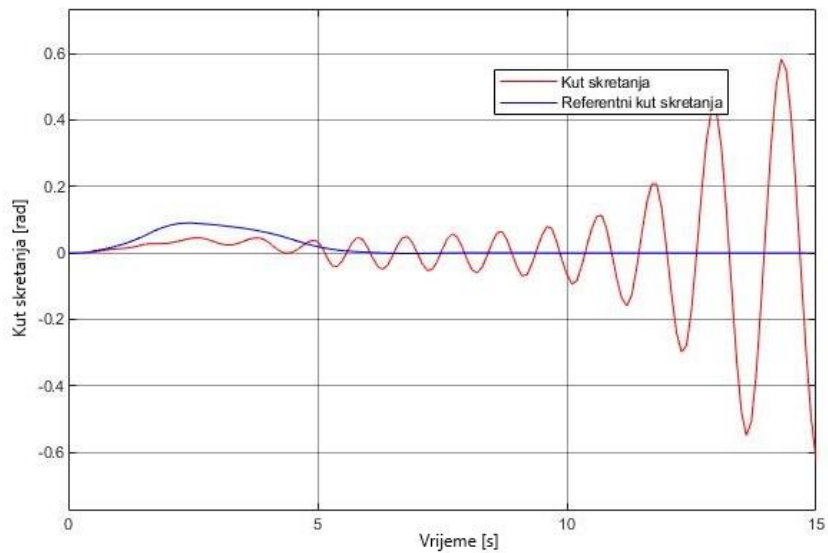
Slika 3.7. Prikaz odziva kuta skretanja u slučaju kada je longitudinalna brzina 15 m/s.

Na slikama se vidi kako dizajnirani regulator radi dobro kada je longitudinalna brzina 15 m/s. Vidi se da lateralni položaj i kut skretanja prate referentnu putanju. To je zato što dinamika vozila koja je korištena u primjeru radi samo za specifične uvjete, a to je kada je longitudinalna brzina 15 m/s. Ukoliko se longitudinalna brzina tokom putovanja promijeni, promijenit će se i dinamika vozila. Dakle, ako se brzina promijeni na 35 m/s, to će uzrokovati degradaciju kakvoće upravljanja, što se može uočiti iz rezultata simulacije danih na slikama 3.6 i 3.7.

Sljedeće slike prikazuju kako se gore projektirani regulator ne može nositi sa longitudinalnom brzinom koja iznosi 35 m/s.



Slika 3.8. Prikaz odziva lateralnog položaja u slučaju kada je longitudinalna brzina 35 m/s.



Slika 3.9. Prikaz odziva kuta skretanja u slučaju kada je longitudinalna brzina 35 m/s.

Na slikama 3.8 i 3.9 se vidi da putanja lateralnog položaja i kuta skretanja ne prati referentnu putanju, tj. došlo je do degradacije kakvoće upravljanja. Kako se to ne bi dogodilo potrebno je dizajnirati adaptivni MPC regulator koji će ažurirati interni model procesa tokom promjene specifičnih uvjeta.

4. ZAKLJUČAK

Autonomna vožnja je zadnjih godina sve zastupljenija. Upravljanje autonomnim automobilom je dosta složen zadatak, stoga se razvojem MPC-a omogućuje optimalno upravljanje. Prvotno je prikazana teorijska osnova MPC-a. Jedna od važnijih komponenata je matematički model procesa koji se koristi za predviđanje budućih izlaza i stanja procesa. Na taj način MPC pronalazi najbolji mogući put za rješavanje problema. U ovom radu korišten je programski paket Matlab/Simulink, pomoću kojeg je simuliran jedostavan primjer. Primjer se odnosi na generiranje manevra promjene trake. U primjeru koji je korišten zadani su specifični uvjeti, što znači da ukoliko se isti promijene doći će do degradacije kakvoće upravljanja. Daljnim razvojem otkriven je adaptivni MPC koji omogućuje da se u svakom koraku s promjenom specifičnih uvjeta pruži novi linearni model sustava, stoga daje točnija predviđanja u novim specifičnim uvjetima.

MPC je zapravo skup pravila za upravljanje, stoga postoje razne formulacije problema i njegovo rješavanje te je ono još uvijek u fazi razvoja i istraživanja.

LITERATURA

1. url: <https://www.mathworks.com/videos/series/understanding-model-predictive-control.html>
Understanding model predictive control, Pristup: 21.04.2021.
2. O. Rodić, Modelska prediktivno upravljanje bočnom silom vozila u manevrima pretjecanja, Diplomski rad, Zagreb, 2019.
3. url:
https://books.google.hr/books?hl=hr&lr=&id=tXZDAAAAQBAJ&oi=fnd&pg=PA13&dq=model+predictive+control&ots=L3Jvnyx_b0&sig=rSM1JHYGhxDPtt9Ek3Yv9VXpgo0&redir_esc=y#v=onepage&q=model%20predictive%20control&f=false Model predictive control, Eduardo F. Camacho, Carlos Bordons Alba, Pristup: 25.06.2021.
4. url: file:///C:/Users/Korisnik/Downloads/Reda_Bouزيد_Vasarhelyi_104.pdf Model Predictive Control for Automated Vehicle Steering, Ahmad Reda, Ahmed Bouzid, József Vásárhelyi, Pristup: 15.08.2021.
5. url: <https://www.sciencedirect.com/science/article/abs/pii/S0967066116300570> Design and evaluation of a model predictive vehicle control algorithm for automated driving using a vehicle traffic simulator, Pristup: 25.06.2021.
6. N. Perić, Automatsko upravljanje, Zagreb, 2005.
7. T. Bariša, Besenzorsko modelsko prediktivno izravno upravljanje strujom sinkronoga vjetrogeneratora sa stalnim magnetima, doktorski rad, Zagreb, 2019.

SAŽETAK

MPC je napredna metoda upravljanja procesima koja služi za predviđanje ponašanja zavisnih varijabli (izlaza) s obzirom na promjenu nezavisnih varijabli (ulaza), uzimajući u obzir dinamiku sustava, ograničenja i ostale parametre te na taj način daje vrijednosti optimalnih nezavisnih varijabli u N narednih koraka. U radu je za primjer odabran jednostavan model vozila te je korištena aplikacija MPC Designer za projektiranje MPC regulatora koji autonomno upravlja vozilom u scenariju manevra promjene trake.

- **Ključne riječi:** modelsko prediktivno upravljanje, autonomno vozilo, ograničenja

ABSTRACT

Introduction to model predictive control

Model predictive control is an advanced method that serves to predict the behavior of dependent variables (outputs) with respect to changing independent variables (inputs), taking into account system dynamics, constraints and other parameters and thus gives us the values of optimal independent variables in N next steps . As part of the work, a simple vehicle model was modeled and the MPC Designer application was used to design an MPC controller that autonomously controls the vehicle in the lane change maneuver scenario.

- **Keywords:** model predictive control, autonomous vehicle, constraint