

Raportul Științific Sintetic

(RSS)

**Control predictiv bazat pe agenți pentru
plutoanele de autovehicule**
*(Multi-agent model predictive control for
vehicle platooning)*

IMPROVING

September 24, 2017

CUPRINS

| | | |
|-------------------|--|-----------|
| 1 | Obiective generale | 7 |
| 2 | Introducere | 8 |
| 2.1 | Definirea problemei | 10 |
| 2.2 | Automatizarea autovehiculelor | 12 |
| 2.2.1 | Vehicule autonome | 13 |
| 2.2.2 | Adaptive Cruise Control | 14 |
| 2.2.3 | Plutoane de autovehicule | 14 |
| 2.2.4 | Comunicații între vehicule | 16 |
| I | Dezvoltarea unui model realist distribuit pentru plutoanele de autovehicule | 18 |
| I.1 | Obiectivele etapei | 18 |
| I.2 (A1.1) | Selectarea celor mai recente modele pentru plutoanele de vehicule și dezvoltarea de noi modele distribuite ce evidențiază cuplajele dintre vehicule | 19 |
| I.2.1 | Stadiul actual referitor la metodele de modelare a dinamicii vehiculelor . . . | 19 |
| I.2.1.1 | Dinamica laterală a vehiculelor | 20 |
| I.2.1.2 | Dinamica longitudinală a vehiculelor | 22 |
| I.2.1.3 | Modelarea forțelor cauciucurilor cu modelul Pacejka | 24 |
| I.2.1.4 | Modelarea dinamicii complete a vehiculului | 25 |
| I.2.1.5 | Implementarea modelului în Matlab/Simulink și rezultate de simulare | 28 |
| I.2.2 | Modelarea lanțului de transmisie a puterii la autovehicule | 33 |
| I.2.2.1 | Model afin | 33 |
| I.2.2.2 | Model afin pe porțiuni | 35 |
| I.2.2.3 | Implementarea modelului în Matlab/Simulink și rezultate de simulare | 38 |
| I.2.3 | Modele roboți mobili | 38 |
| I.2.3.1 | Modelarea roboților mobili mOway | 38 |
| I.2.3.2 | Modelarea roboților mobili SRV | 39 |
| I.2.4 | Modele ale plutoanelor de autovehicule | 40 |
| I.2.4.1 | Configurația unui pluton și modelarea dinamicii vehiculului | 40 |
| I.2.4.2 | Modelarea unui pluton de vehicule cu ACC centralizat | 42 |
| I.3 (A1.2) | Modelarea întârzierilor variabile în timp din rețea folosind aproximări politopice sau considerându-le ca perturbații pentru a le exprima ca partiții poliedrale ale spațiului stărilor | 44 |
| I.3.1 | Modelarea întârzierilor folosind aproximări politopice | 44 |
| I.3.2 | Modelarea întârzierilor de intrare și de ieșire ca perturbații | 45 |
| I.3.2.1 | Întârzieri pe calea directă | 45 |
| I.3.2.2 | Întârzieri pe calea de reacție | 45 |

| | |
|---|-----------|
| I.4 (A1.3) Modelarea restricțiilor hardware ca partiții poliedrale ale spațiului stărilor | 46 |
| I.4.1 Implementarea reguletoarelor predictive în timp real | 46 |
| I.4.1.1 Arhitectura hardware folosită | 47 |
| I.4.1.2 Particularitățile constructive ale roboților mobili utilizați în sisteme multi-agent sau de tip pluton | 50 |
| I.4.2 Restricții hardware | 51 |
| I.5 (A1.4) Integrarea optimă a acestor modele într-un model distribuit realist pentru PV și (A1.5) Analiza modelului complet din punctul de vedere al strategiilor de control | 52 |
| I.5.1 Metodologii de control | 52 |
| I.5.1.1 Reguletoare PID | 52 |
| I.5.1.2 Metoda Sliding Mode Control (SMC) | 56 |
| I.5.1.3 Metoda Model Predictive Control (MPC) | 59 |
| I.5.2 Modele de vehicule utilizate în controlul predictiv distribuit | 60 |
| I.6 Concluzii | 65 |
| II Dezvoltarea strategiilor de control predictiv bazate pe agenți | 67 |
| II.1 Obiectivele etapei | 67 |
| II.2 (A2.1) Dezvoltarea de strategii predictive de control folosind agenți bazate pe PLF-uri utilizând modelul distribuit realist dezvoltat în Etapa 1 și considerând perturbațiile din trafic | 68 |
| II.2.1 Dezvoltarea unui algoritm de control predictiv distribuit non-cooperativ pentru urmărirea referinței | 68 |
| II.2.1.1 Introducere | 68 |
| II.2.1.2 Modelul <i>de viteză</i> folosit în proiectarea reguletoarelor predictive | 69 |
| II.2.1.3 Problema de optimizare non-cooperativă DMPC | 70 |
| II.2.1.4 Pluton de vehicule descrise prin model dublu integrator | 73 |
| II.2.1.5 Pluton de vehicule descrise prin model realist | 82 |
| II.2.2 Controlul vitezei de croazieră și a distanței dintre vehiculele unui pluton folosind un algoritm de control predictiv distribuit | 84 |
| II.2.2.1 Introducere | 85 |
| II.2.2.2 Modelarea vehiculelor și formularea problemei de control | 86 |
| II.2.2.3 Implementarea algoritmului DMPC | 88 |
| II.2.2.4 Rezultate obținute prin simulare | 91 |
| II.2.2.5 Concluzii | 93 |
| II.2.3 Algoritm de control predictiv distribuit coalițional cu formulare min-max | 95 |
| II.2.3.1 Introducere | 95 |
| II.2.3.2 Algoritmul MIN-MAX descentralizat | 96 |
| II.2.3.3 Algoritmul MIN-MAX distribuit | 97 |
| II.2.3.4 Algoritmul MIN-MAX DMPC coalițional | 99 |

| | | |
|----------|--|-----|
| II.2.3.5 | Rezultate | 101 |
| II.2.3.6 | Concluzii | 105 |
| II.2.4 | Sisteme cyber fizice - proiectarea orientată a controlului cooperativ pentru plutoane de vehicule | 106 |
| II.2.4.1 | Modelul vehiculului | 107 |
| II.2.4.2 | Controlul vehiculului pentru deplasarea în pluton | 108 |
| II.2.4.3 | Metoda de control pentru liderul plutonului | 108 |
| II.2.4.4 | Algoritmul de control pentru urmăritori | 110 |
| II.2.4.5 | Proiectarea unui pluton cu CACC | 112 |
| II.2.4.6 | Studiu de caz | 113 |
| II.2.5 | Controlul cooperativ al unui pluton de vehicule pentru creșterea capacității arte-urilor de circulație urbane | 116 |
| II.2.5.1 | Modelarea vehiculelor din pluton | 117 |
| II.2.5.2 | Sistemul CACC pentru controlul longitudinal al unui pluton de vehicule | 119 |
| II.2.5.3 | Studiu de caz | 122 |
| II.2.6 | Controlul predictiv al dinamicii vehiculelor autonome | 126 |
| II.2.6.1 | Modelarea matematică a dinamicii mașinii | 126 |
| II.2.6.2 | Controlul predictiv neliniar al dinamicii laterale a vehiculului | 129 |
| II.2.6.3 | Controlul predictiv neliniar al dinamicii laterale a vehiculelor cu funcții de stabilitate Lyapunov | 131 |
| II.2.6.4 | Controlul predictiv neliniar multivariat al dinamicii vehiculelor | 134 |
| II.2.6.5 | Controlul predictiv centralizat al dinamicii vehiculelor autonome folosind modele liniare | 137 |
| II.2.7 | Planificarea optimă a traiectoriilor pentru vehicule autonome | 139 |
| II.2.8 | Controlul predictiv al lanțului de transmisie a puterii la autovehicule în rețea bazat pe restricții probabilistice | 147 |
| II.2.8.1 | Formularea problemei | 150 |
| II.2.8.2 | Modelarea întârzierilor introduse de rețeaua de comunicații | 153 |
| II.2.8.3 | Configurația sistemului de control în rețea | 153 |
| II.2.8.4 | Modelarea lanțului de transmisie al vehiculului | 155 |
| II.2.8.5 | Rezultate de simulare | 159 |
| II.2.9 | Control predictiv robust pentru sistemele controlate în rețea considerând pierderile de pachete de date ca perturbații | 161 |
| II.2.9.1 | Modelarea pierderilor de pachete de date ca perturbații | 164 |
| II.2.9.2 | Metodologia reglării predictive robuste | 167 |
| II.2.9.3 | Modelul sistemului de servodirecție electric | 169 |
| II.2.9.4 | Rezultate de simulare | 170 |

II.3(A2.2) Dezvoltarea de emulatoare în Matlab pentru plutoanele de vehicule cooperante **175**

| | | |
|--------|--|-----|
| II.3.1 | Emulator Matlab Simulink pentru un pluton de 5 vehicule modelate cu dublu integrator | 175 |
|--------|--|-----|

| | |
|---|------------|
| II.3.2 Emulator Matlab Simulink pentru un pluton de 5 vehicule modelate prin descrierea lanțului de transmisie a puterii | 179 |
| II.3.3 Proiectarea unui sistem CACC cu reglarea după stare | 180 |
| II.3.3.1 Modelul de mișcare longitudinală a vehiculului | 181 |
| II.3.3.2 Formularea problemei de control a CACC | 182 |
| II.3.3.3 Proiectarea sistemului CC pentru vehiculul lider | 184 |
| II.3.3.4 Proiectarea sistemului de control pentru vehiculele următoare | 184 |
| II.3.3.5 Simulator Matlab pentru un pluton cu CACC | 186 |
| II.3.4 Proiectarea unui pluton cu controlul predictiv bazat pe modelul în spațiul stărilor | 190 |
| II.3.4.1 Controlul predictiv fără restricții | 191 |
| II.3.4.2 Controlul predictiv cu restricții | 193 |
| II.3.4.3 Simulator Matlab pentru un pluton de vehicule cu MPC cu restricții | 194 |
| II.4(A2.3) Verificarea stabilității plutonului de vehicule cooperante bazate pe multiagenți | 197 |
| II.5(A2.4) Analiza strategiei de control predictive bazate pe agenți | 201 |
| III Implementarea în timp real și analiza performanțelor | 203 |
| III.Obiectivele etapei | 203 |
| III.2(A3.1) Dezvoltarea programelor de timp real pentru implementarea algoritmilor de control predictivi distribuiți bazați pe agenți în sisteme încorporate | 204 |
| III.2.1 Descrierea mașinii prototip pentru studiul plutoanelor de vehicule | 204 |
| III.2.1.1 Proiectarea și realizarea părții mecanice a mașinii prototip | 207 |
| III.2.1.2 Prezentarea senzorilor și dispozitivelor ce compun sistemul încorporat al mașinii prototip | 216 |
| III.2.1.3 Proiectarea și realizarea plăcii principale cu microcontroller | 223 |
| III.2.1.4 Prezentarea aplicației încorporate destinate identificării | 234 |
| III.2.1.5 Prezentarea aplicației încorporate bazate pe FreeRTOS | 240 |
| III.2.2 Implementarea algoritmilor de control în sistemul încorporat | 250 |
| III.2.2.1 Implementarea algoritmului de control PID | 250 |
| III.2.2.2 Implementarea algoritmului de control CACC-GPC | 251 |
| III.2.2.3 Implementarea algoritmului de control DMPC | 253 |
| III.2.3 Algoritm DMPC în formulare cu model de viteză | 254 |
| III.3(A3.2) Testarea strategiei de control predictiv distribuit pe un pluton de roboți mobili | 256 |
| III.3.1 Control predictiv descentralizat fără comunicații pentru o formație de roboți mobili | 256 |
| III.3.1.1 Introducere | 256 |
| III.3.1.2 Arhitectura de control | 257 |
| III.3.1.3 Proiectarea regulatorului predictiv în spațiul stărilor (SS-MPC) | 259 |

| | | |
|--------------|--|------------|
| III.3.1.4 | Strategia EPSAC | 261 |
| III.3.1.5 | Roboții mobili mOway | 262 |
| III.3.1.6 | Rezultate obținute | 263 |
| III.3.1.7 | Stabilitate | 264 |
| III.3.1.8 | Rezultate obținute în simulare | 266 |
| III.3.1.9 | Rezultate experimentale | 266 |
| III.3.1.10 | Remarci experimentale | 268 |
| III.3.1.11 | Evaluarea performanței | 269 |
| III.3.1.12 | Concluzii | 269 |
| III.3.2 | Utilizarea comunicațiilor wireless pentru colectarea datelor în plutoanele de vehicule | 272 |
| III.3.2.1 | Descrierea sistemului | 274 |
| III.3.2.2 | Comunicații wireless | 275 |
| III.3.2.3 | Proiectarea regulatorului predictiv | 278 |
| III.3.2.4 | Rezultate de simulare și experimentale | 281 |
| III.3.2.5 | Remarci asupra rezultatelor experimentale | 282 |
| III.3.2.6 | Remarci asupra comunicației | 282 |
| III.4 | Analiza performanțelor în timp real a algoritmilor predictivi bazați pe agenți și compararea cu soluțiile existente pentru plutoanele de vehicule | 283 |
| III.4.1 | Descrierea performanțelor în timp real a algoritmului de tip PID | 283 |
| III.4.2 | Descrierea performanțelor în timp real a algoritmului de tip CACC-GPC | 286 |
| III.4.3 | Descrierea performanțelor în timp real a algoritmului de tip DMPC | 286 |
| | Bibliografie | 310 |

1 Obiective generale

Principalul obiectiv al acestui proiect este de a crea un cadru general pentru modelarea și controlul vehiculelor cooperante ce se deplasează în plutoane considerate ca sisteme multiagent. Metodologia va include imperfecțiunile introduse de comunicația fără fir, a restricțiilor hardware/computaționale și a perturbațiilor din trafic chiar din faza de proiectare a regulatorului pentru a îmbunătăți siguranța, pentru a reduce consumul de combustibil, pentru a crește capacitatea drumurilor și pentru a reduce emisiile.

Îndeplinirea *obiectivului principal* al proiectului se bazează pe un set de obiective specifice care vizează rezolvarea problemelor apărute în deplasarea vehiculelor în plutoane, din cauza rețelelor de comunicații, a restricțiilor hardware și a celor fizice și a perturbațiilor din trafic:

- **Obiectiv 1 (Etapa I):** cadru de modelare pentru plutoanele de vehicule distribuite care este capabil să includă cuplajul dintre vehicule, comunicația fără fir și restricțiile hardware/ computaționale: pornind de la tehnicile existente de modelare a întârzierilor variabile în timp folosind aproximări poliedrice sau prin considerarea lor ca perturbații, efectul întârzierilor va fi reprezentat în partiții poliedrale ale spațiului stărilor; mai mult, restricțiile hardware/ computaționale vor fi reprezentate în partiții poliedrale ale spațiului stărilor; în plus cele două metode vor fi integrate astfel încât să se obțină un model realist al plutonului de vehicule care să includă cuplajele dintre vehicule, întârzierile variabile în timp și restricțiile hardware/computaționale;
- **Obiectiv 2 (Etapa II):** dezvoltarea unui cadru pentru proiectarea de regulatoare predictive distribuite bazate pe agenți capabile să utilizeze cadrul de modelare de la primul obiectiv și să ia în considerare condițiile meteorologice (ex., drumuri uscate, umede sau alunecoase) și configurația drumurilor (ex., rampă sau pantă); rezultă legi de reglare predictive bazate pe PLF-uri ce vor fi testate într-un mediul de simulare dezvoltat în Matlab pentru plutoane de vehicule cooperante;
- **Obiectiv 3 (Etapa III):** dezvoltarea de programe de timp real pentru implementarea algoritmilor de control predictivi distribuiți multiagent pe sisteme încorporate ce vor fi testate experimental în laborator folosind un pluton de vehicule (pluton de roboți mobili); va fi efectuată analiza performanțelor și compararea cu soluțiile existente în literatura de specialitate.

2 Introducere

Accidentele rutiere rămân o cauză majoră a deceselor și rănilor în Europa în ciuda îmbunătățirilor considerabile din ultimul deceniu. Comisia Europeană recunoaște rolul important al sistemelor de transport inteligente pentru creșterea siguranței pe drumurile europene. [1]. Mai mult, transportul rutier utilizează aproximativ 27% din totalul energiei consumate în Uniunea Europeană [2]. Fluxul crescut al traficului pe autostrăzile existente și pe drumurile din orașe are multe dezavantaje, ex., creșterea riscului de accidente, consum mai mare de combustibil, creșterea poluării și a uzurii părților mecanice ale vehiculului, stres în timpul șofatului și disconfort pentru pasageri [3, 4].

O soluție la aceste probleme este dată de Sistemele Automate de Autostrăzi (SAA), care sunt capabile să asigure o coordonare sigură și eficientă a vehiculelor. Acest lucru se face prin înlocuirea conducătorilor auto de către regulatoare pentru plutoanele de autovehicule, care impune vehiculelor să se urmeze reciproc și să mențină distanțe de siguranță între ele [5–7]. Mai mult decât atât, a fost deja dovedit în [7, 8], că prin introducerea comunicațiilor între autovehicule, performanțele plutoanelor rezultate sunt mult mai mari decât cele obținute fără schimb de informații între vehicule. Unele dintre marile avantaje ale SAA sunt următoarele: reducerea riscului de accidente, o mai mare economie de combustibil datorită rezistenței aerului redusă, eficiență mai mare a vehiculelor și creșterea capacității autostrăzilor [9]. Cu toate acestea, acest lucru vine cu alte probleme care trebuie luate în considerare: nicio coliziune nu trebuie să se producă în caz de frânare de urgență și fiecare vehicul dintr-un pluton trebuie să aibă o dinamică stabilă, precum și întregul pluton. Această proprietate este denumită stabilitate a șirurilor (stabilitate pluton), care prevede ca în cazul în care are loc o deviere de la distanța dorită dintre lider și primul urmăritor, această eroare ar trebui să scadă spre spatele plutonului [10].

Plutoanele de autovehicule pot fi soluția acestor probleme stringente prin impunerea ca vehiculele să se urmeze unele pe altele și să păstreze o distanță sigură între ele (Naus et al., 2010; Zhao et al., 2010; Oncu et al., 2012). Mai mult, a fost deja demonstrat în (Oncu et al., 2012; Jia et al., 2014) că prin includerea comunicației între vehicule performanțele plutoanelor rezultate sunt mult mai bune decât cele obținute fără comunicație între vehicule. Totuși, avantajele acestei soluții depind de asigurarea stabilității plutonului de vehicule (Swaroop and Hedrick, 1996).

Problema este de a sintetiza regulatoare încorporate de timp real (văzute ca agenți care negociază pentru a obține performanțe optime) pentru plutoanele de vehicule luând în considerare restricțiile platformei de implementare, ale canalului de comunicații și perturbațiile din trafic, restricții ce nu sunt luate în calcul de metodele existente de sinteză a regulatoarelor. Mai mult, modelul plutonului de vehicule (PV) trebuie să fie suficient de complex pentru a surprinde dinamica vehiculelor reale și a cuplajelor dintre ele, dar destul de simplu pentru proiectarea regulatoarelor.

Într-un pluton de vehicule negocierea pentru a obține performanțe optime este realizată prin rețele de comunicații fără fir, iar folosirea lor implică necesitatea considerării efectelor imperfecțiunilor introduse de rețea: întârzieri variabile în timp, pierderi de pachete de date și perioade de eșantionare variabile (Tipsuwan and Chow, 2003). Mai

mult, metodele de control avansate, cum ar fi controlul predictiv bazat pe agenți care ar trebui să țină seama de cuplajele dintre vehicule, se concentrează în primul rând pe obținerea performanțelor și apoi pe satisfacerea restricțiilor hardware/fizice și rejectarea perturbațiilor. Această abordare în doi pași conduce la regulatoare ineficiente în timp real sau la o diminuare a performanțelor. Astfel, este nevoie de o translație cantitativă a specificațiilor hardware în restricții compatibile cu sinteza regulatorului.

Autonomia unui vehicul este definită ca fiind capacitatea vehiculului de a lua decizii și de a acționa fără intervenție umană. Ca atare, se poate spune că în zilele noastre vehiculele sunt parțial autonome, fiind echipate cu sisteme de control a vitezei de croazieră (cruise control - CC) [11], pentru a menține vehiculul pe o anumită bandă de circulație [12], pentru a evita blocarea frânelor în caz de urgență, pentru a parca autovehiculul în mod automat. În viitorul apropiat, se anticipează dezvoltarea sistemelor de asistență avansate, adică, sisteme care pot detecta posibilitatea iminentă de producere a unei coliziuni în intersecții sau schimbarea neintenționată a unei benzi. În zilele noastre, comportamentul autonom al vehiculelor este studiat atât în industrie, cât și în mediul academic, ceea ce a condus la dezvoltarea unor concepte promițătoare, de exemplu, formarea plutoanelor de autovehicule [11, 13, 14], urmărirea sau schimbarea benzii de circulație [12, 15], oprirea de urgență și evitarea obstacolelor [11, 12, 16].

Considerând vehiculele cu sisteme de control automatizate, este posibil să se creeze un pluton de autovehicule, care se poate deplasa automat prin urmărirea unui vehicul-lider, care poate fi controlat în mod automat prin utilizarea de comunicații cu infrastructura rutieră sau manual de către un conducător auto. Plutonul de vehicule circulă pe o singură bandă, prin utilizarea unei strategii de control longitudinal pentru fiecare vehicul, dar pot fi, de asemenea, luate în considerare cazurile în care un vehicul trebuie să intre sau să iasă din pluton, care implică controlul lateral al vehiculului.

Literatura de specialitate disponibilă propune două abordări referitoare la plutoanele de autovehicule: fie luând în considerare o distanță fixă pentru a fi menținută între vehiculele din pluton [5–7, 13, 17–20] sau luând în considerare o distanță care variază în funcție de viteza vehiculului în pluton [5, 6, 21, 22].

Soluțiile existente pentru deplasarea în plutoane a vehiculelor includ comunicațiile V2V între vehicule [10, 23–29], dar doar unele dintre ele [24, 27, 29] analizează efectele restricțiilor și nici una din ele nu consideră negocierea dintre vehicule. Fezabilitatea rețelelor de comunicații fără fir trebuie garantată pentru negocierea în timp real între vehiculele din pluton cu scopul obținerii performanțelor optime. Mai mult, doar în [10] și citeNemeth2011 perturbațiile din trafic sunt considerate într-o anumită măsură și nici una din lucrările de mai sus nu consideră restricțiile hardware.

Particularitățile specifice ale plutoanelor de vehicule, adică structura de timp înlănțuit a subsistemelor implicate (vehicule), le face potrivite pentru utilizarea strategiei de control al predictiv distribuit bazat pe model (DMPC). DMPC presupune împărțirea unui sistem la scară largă (întregul pluton) în sub-sisteme individuale (vehicule) cuplate în mod dinamic prin intrări sau stări [30]. Ideea este de a controla fiecare subsistem (vehicul), de către un agent local (regulator), utilizând informațiile disponibile la nivel local și cunoștințele relevante obținute prin comunicare între agenții vecini, adică, toți agenții care sunt cuplați.



Figura 1: Vehicule cooperante în pluton.

2.1 Definirea problemei

Reprezentarea generală a plutoanelor formate din n vehicule este ilustrată în Fig. 1 unde vehiculele (V_i) cu agenți de control (A_i) schimbă informații și negociază folosind comunicațiile vehicul-la-vehicul (V2V) prin rețele fără fir pentru a-și îmbunătăți performanțele și pentru a asigura stabilitatea întregului pluton. Poziția lor absolută este detectată folosind un senzor de poziționare global (GPS) montat pe autovehicul și distanța relativă d dintre vehiculul V_i și vehiculul V_{i-1} este determinată folosind un dispozitiv radar. Deplasarea în plutoane presupune ca fiecare vehicul din pluton să-și controleze viteza și distanța relativă față de vehiculul din față (d) prin negocierea cu ceilalți agenți de control din pluton pentru a îmbunătăți siguranța rutieră, a reduce consumul de combustibil, a crește capacitatea drumurilor și a reduce emisiile poluante.

Obiectivele urmărite pentru proiectarea sistemelor de control într-un pluton de vehicule sunt stabilite pe baza următoarelor cerințe [25]:

1. ținând seama de caracteristicile vehiculului, starea suprafeței drumului, precum și situațiile reale din trafic, strategia corectă între vehicule trebuie să fie selectate de către conducătorul auto;
2. starea de stabilitate corespunzătoare eroarii de distanță ar trebui să fie egală cu 0 pentru toate vehiculele din pluton, astfel încât să se asigure capacitatea de urmărire a vehiculelor;
3. atunci când vehiculul lider accelerează, vehiculul urmăritor ar trebui să accelereze înainte de a detecta că cel mai apropiat vehicul accelerează, pentru a evita creșterea distanței între vehicule;
4. atunci când vehiculul lider frânează, vehiculul urmăritor trebuie, de asemenea, să decelereze înainte de a detecta că vehiculul anterior decelerează, pentru a evita coliziunea.

Obiectivele (2), (3) și (4) pot fi utilizate ca abilitatea de urmărire a vehiculului. Pentru a satisface obiectivele (3) și (4), viteza dorită și accelerația vehiculului trebuie să fie controlate pe baza informațiilor primite de la liderul plutonului și / sau de la cel mai apropiat vehicul din față acestuia.

Mai mult decât atât, stabilitatea întregului pluton de vehicule este unul dintre punctele majore care trebuie luate în considerare atunci când sunt dezvoltate regulatoarele longitudinale [10], pentru a se evita efectele secundare negative asupra fluxului sau siguranța traficului. Pentru a asigura stabilitatea unui pluton de vehicule trebuie să fie îndeplinite în fiecare moment de timp două condiții [10, 31]:

- eroarea tranzitorie nu ar trebui să amplifice, adică eroarea de distanță dintre vehicule trebuie să fie atenuată, pornind de la lider spre ultimul vehicul din pluton;
- valorile absolute ale accelerațiilor ar trebui să scadă spre partea din spate a plutonului.

Din punct de vedere tehnologic, deplasarea vehiculelor în plutoane permite utilizarea strategiilor avansate de control, ex., controlul predictiv (MPC) distribuit bazat pe sisteme multiagent, care poate asigura performanțele vehiculelor și stabilitatea plutonului, luând în considerare imperfecțiunile rețelei de comunicații fără fir și restricțiile hardware și adaptează regulatorul la perturbațiile din trafic (ex., condițiile meteorologice și configurația drumului).

Referitor la punctul de vedere socio-economic, controlul plutoanelor de vehicule cooperante utilizând un algoritm distribuit duce la scăderea riscului de accidente, minimizarea consumului de combustibil, reducerea emisiilor poluante și reducerea costurile de implementare/întreținere.

Problema care se pune este de a sintetiza regulatoare încorporate în timp real (privite ca agenți care negociază pentru obținerea unor performanțe optime) pentru plutoanele de vehicule, luându-se în considerare restricțiile derivate din platforma de implementare, canalul de comunicare și perturbațiile din trafic, restricții care nu sunt luate în considerare de metodele existente de sinteză ale regulatoarelor. Mai mult decât atât, modelul plutonului de vehicule ar trebui să fie suficient de complex pentru a se potrivi caracteristicilor dinamice ale vehiculelor reale și cuplajelor dintre ele, dar destul de simple pentru a fi utilizate pentru proiectarea regulatoarelor.

Complexitatea crescândă a cerințelor de siguranță și confort, păstrând în același timp sub control costurile de produs, reprezintă o provocare majoră în industria auto. Ca o consecință, punerea în aplicare în timp real a algoritmilor de control trebuie să îndeplinească cerințele hardware stricte și limitările impuse [32]. Acest aspect nu este luat în considerare de către soluțiile existente în faza de proiectare a regulatorului, ceea ce duce la sarcini care durează prea mult, iar unele funcționalități nu sunt efectuate. Astfel, o reconfigurare mai flexibilă a sarcinilor și a mesajelor în conformitate cu reconfigurarea realizată în ceea ce privește mărimea de control ar fi necesară.

Într-un pluton de vehicule, negocierea pentru a obține performanțe optime se realizează prin intermediul rețelelor de comunicații fără fir. Utilizarea lor presupune proiectarea unor regulatoare care să facă față efectelor imperfecțiunilor induse de rețea: întâzieri variabile în timp, pierderi de pachete de date și perioade de eșantionare variabile [33–35]. Mai mult decât atât, metodele avansate de control, cum ar fi controlul predictiv (MPC), care ar trebui să ia în considerare cuplajele dintre vehicule, se axează în primul rând pe obținerea performanțelor și numai după aceea pe restricțiile hardware / ale procesului și pe rejectarea perturbațiilor. Această abordare în două etape, fie conduce la regulatoare nefezabile în timp real sau pierdere de performanță. Astfel, este nevoie de o traducere cantitativă a specificațiilor hardware compatibile cu restricții în sinteza regulatoarelor pentru a permite dezvoltarea metode de sinteză cu restricții hardware.

Până în prezent, imperfecțiunile de comunicare [36, 37] și hardware / constrângeri de calcul [38] au fost tratate separat și, în plus, acestea din urmă nu au fost luate în considerare în faza de proiectare a regulatorului, ci într-un mod ad-hoc, prin metode de încercare

și eroare. Cu toate acestea, mai multe încercări au fost făcute pentru a pune în aplicare regulatoare predictive cu restricții hardware pe dispozitive dedicate, de exemplu, bazate pe FPGA [39] sau bazate pe microcontroler [38], dar timpii de calcul sunt mult mai mari decât ceea ce este necesar pentru plutoanele de vehicule și implementările nu țin cont de restricțiile comunicației fără fir.

2.2 Automatizarea autovehiculelor

Progresele tehnologice creează o continuă dezvoltare de la vehiculele convenționale, total controlate de om, la vehiculele autonome sau automate (AV), care se pot conduce singure parțial sau total și care într-un final ar putea să nu mai necesite deloc prezența unui șofer. Există tehnologii care activează un vehicul să asiste și să ia decizii pentru un conducător uman. Astfel de tehnologii includ sisteme de avertizare a pericolului de coliziune (CW/CA), sisteme adaptive de control a vitezei de croazieră (ACC), sisteme de menținere a benzii de deplasare (lane keeping systems), și tehnologia de parcare automată.

Există o ierarhie pe cinci nivele care poate ajuta la înțelegerea mai ușoară a acestei dezvoltări continue și a legăturii dintre tehnologii [40]:

- Nivelul 0 (fără automatizare): șoferul este singurul care controlează funcțiile primare ale vehiculului (frâne, direcție și accelerație) tot timpul, și doar acesta este responsabil pentru monitorizarea drumului și a operațiunilor de deplasare sigure.
- Nivelul 1 (automatizare specifică la nivel de funcție): La acest nivel automatizarea implică una sau mai multe funcții specifice de control. Conducătorul auto deține controlul total, și este singurul responsabil pentru operațiunile de siguranță, dar poate alege să cedeze o autoritate limitată către o funcție primară (precum în cazul ACC).
- Nivelul 2 (automatizare bazată pe funcții combinate): Acest nivel implică automatizarea a cel puțin două funcții primare de control proiectate să lucreze împreună pentru a elibera șoferul de controlul acelor funcții. În acest caz vehiculul este capabil să împartă autoritatea când șoferul cedează controlul în condiții limitate de conducere. Tot șoferul este responsabil pentru acționarea sistemelor de siguranță ale autovehiculului.
- Nivelul 3 (automatizare pentru auto-conducere limitată): Vehiculele la acest nivel de automatizare permit șoferului să cedeze în totalitate controlul tuturor funcțiilor critice din punctul de vedere al siguranței în cazul unor anumite condiții de trafic sau de mediu. Șoferul trebuie să fie disponibil pentru controlul ocazional, dar cu un timp de tranziție suficient de confortabil.
- Nivelul 4 (automatizare pentru o conducere pur automată): Vehiculul este proiectat să efectueze toate funcțiile critice pentru siguranță și să monitorizeze condițiile drumului pentru o întreagă călătorie.

Tipul și amplitudinea potențialelor beneficii ale tehnologiilor pentru vehiculele autonome (AV) depind de nivelul de automatizare care este atins. Actualul proiect urmărește realizarea unui pluton de autovehicule care să aibă un grad de automatizare care să se încadreze la nivelul 3.

2.2.1 Vehicule autonome

”Autonomia” unui vehicul este definită drept capacitatea vehiculului de a lua decizii și a acționa fără intervenția omului. Drept urmare, se poate spune că vehiculele actuale sunt ”parțial autonome”, fiind echipate cu sisteme pentru: controlul vitezei de croazieră (eng. cruise control) [11], evitarea ieșirii de pe banda de rulare (eng. lane departure avoidance) [12, 41], asistență la frânarea de urgență (eng. antilock brake systems (ABS)), parcare automată (eng. autonomous parking). În viitorul apropiat, este anticipată dezvoltarea sistemelor avansate de asistență care pot detecta posibilitatea producerilor de coliziuni în intersecții și schimbarea neintenționată a benzii. În prezent, comportamentul autonom al vehiculelor este studiat atât în industrie, cât și în mediul academic, ceea ce a dus la apariția unor concepte promițătoare, cum ar fi urmărirea vehiculului din față (eng. convoying, platooning) [11, 14, 42], urmărirea sau schimbarea benzii (eng. lane tracking, lane change) [12, 15, 41], frânarea de urgență (eng. emergency stopping), evitarea obstacolelor (eng. obstacle avoidance) [11, 12, 16, 41].

Principalele componente necesare implementării funcționalităților amintite sunt într-o continuă dezvoltare, crescând în performanțe, permițând astfel tranziția spre dezvoltarea vehiculelor autonome. Aceste componente sunt:

- *Senzorii* - Pentru a putea controla un vehicul în mod automat, algoritmul de control trebuie să cunoască starea vehiculului, deci sunt necesare măsurători ale vitezei vehiculului și a direcției de deplasare, care pot fi realizate folosind senzori de viteză unghiulară a roților, senzori de măsurare a unghiului de virare sau GPS. În plus, algoritmul de control trebuie să cunoască poziția vehiculului relativ la alte vehicule și obstacole. În acest scop pot fi folosite, sisteme radar, lidar, laser sau senzori cu ultrasunete. Nu în ultimul rând, vehiculul trebuie să-și cunoască poziția pe carosabil, ceea ce poate fi realizat utilizând sisteme de vedere artificială sau GPS.
- *Buclele de reglare* - Controlul automat al vehiculelor presupune interfațarea calculatoarelor cu vehiculul prin intermediul elementelor de execuție. Buclele specifice de control sunt: bucla de control a vitezei vehiculului (controlul accelerației) [11], bucla de control a direcției [43], bucla de control a frânării. În cazul în care vehiculul nu este echipat cu transmisie automată, este necesară și o buclă de control a cutiei de viteze.
- *Rețelele de comunicație* - Comunicatiile pot furniza informații generale referitoare la trafic și la mediul înconjurător, precum și atenționări în timp real. Conceptele de comunicații precum vehicul la vehicul (eng. vehicle-to-vehicle (V2V)) și vehicul la infrastructură / infrastructură la vehicul (eng. vehicle-to-infrastructure (V2I), infrastructure-to-vehicle (I2V)) sunt într-o continuă dezvoltare.
- *Algoritmi inteligenți de control* - Algoritmii de control trebuie să simuleze cât mai fiabil comportamentul unui conducător uman. Astfel, acești algoritmi trebuie să asigure urmărirea traiectoriei impuse și siguranța pasagerilor și a tuturor participanților la trafic.

În anticiparea acestei dezvoltări, în prezent marile companii testează prototipuri de vehicule complet autonome, estimându-se că în următorii zece ani mașinile complet autonome

vor intra în producția de serie. Companii precum BMW, Audi, Mercedes-Benz, Ford, GM, Toyota, Nissan și Volvo au prezentat deja concepte de vehicule autonome, în încercarea de a transforma cercetările în soluții fezabile pentru producție. Prototipurile de vehicule autonome produse de Google (eng. Google Car), au devenit apariții familiare pe străzile din California, iar compania continuă să cerceteze și să producă software în vederea creșterii performanțelor și fiabilității acestor vehicule.

Pe termen lung, beneficiile vehiculelor complet autonome sunt ușor de anticipat, putând fi enumerate: eliminarea posibilelor erori umane va duce la creșterea considerabilă a siguranței în trafic, spațiile de parcare nu vor mai fi o problemă deoarece vehiculul va putea căuta locul de parcare și se va putea parca automat, va oferi posibilitatea pasagerilor de a folosi timpul petrecut în vehicul în mod mai productiv.

2.2.2 Adaptive Cruise Control

Sistemul adaptiv de control al vitezei de croazieră (Adaptive Cruise Control ACC) reprezintă o extensie a variantei clasice a acestui sistem (Cruise Control CC), care este o funcționalitate răspândită în cadrul vehiculelor moderne. Rolul CC este acela de a asigura o deplasare a vehiculului cu o viteză constantă stabilită de către șofer prin captarea acesteia în timpul deplasării autovehiculului. În momentul în care este necesară o scădere sau o creștere a vitezei vehiculul realizează automat frânarea sau accelerarea pentru a ajunge la viteza stabilită. Deoarece s-a dorit ca acest sistem să asigure și o mai bună siguranță în deplasarea vehiculului, a fost realizată o îmbunătățire pentru acesta. Această modificare constă în setarea vitezei de croazieră în funcție de viteza autovehiculului din față prin sistemele ACC. Acest lucru făcând astfel posibilă urmărirea în mod automat a vehiculului precedent. Detecția vehiculului din față este realizată prin utilizarea unui senzor de tip radar, a cărui funcționalitate ar putea fi extinsă de o cameră video care ar face automobilul să detecteze mai ușor vehiculul precedent.

Sistemele ACC sunt compuse din două părți: o parte independentă de vehicul și alta dependentă de acesta. Componenta independentă determină un profil al vehiculului în ceea ce privește accelerația și decelerația. Partea dependentă de vehicul asigură urmărirea acestui profil prin controlul sistemelor de accelerație, respectiv de frână [5].

Din dorința producătorilor de mașini de a realiza vehicule cât mai inteligente care să asigure în mod autonom o siguranță sporită a șoferului și a pasagerilor, la aceste sisteme ACC au fost adăugate sisteme de evitare a coliziunilor (collision avoidance CA). Prin implementarea acestora la nivelul autoturismelor s-a ajuns și la un control lateral al acestora, nu doar la unul longitudinal asigurat de ACC. Sistemele pentru CA permite vehiculului să trimită comenzi către sistemele de direcție și de frână pentru a fi acționate automat conducând astfel la evitarea unui obstacol care ar putea periclita siguranța în deplasare.

2.2.3 Plutoane de autovehicule

Un pluton reprezintă o grupare de vehicule care se deplasează împreună cât timp mențin o anumită configurație geometrică fără ajutorul altor materiale de cuplare. Pentru a defini diferitele configurații pe care un pluton le poate avea este nevoie de o definiție a termenilor de distanță laterală și de distanță longitudinală.

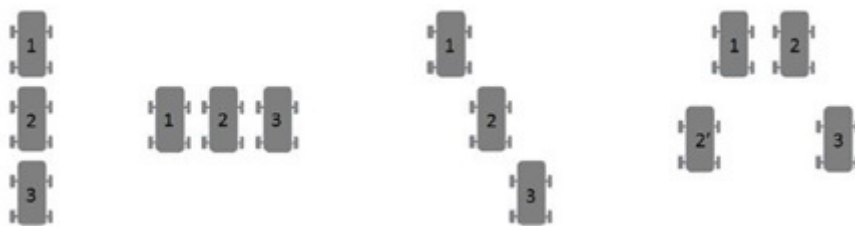


Figura 2: Configurații de plutoane. a) coloană; b) linie; c) eșalon; d) pană.

Distanța laterală este spațiul orizontal dintre două vehicule alăturate, iar distanța longitudinală reprezintă spațiul vertical dintre acestea. Configurarea unui pluton depinde de definirea ambilor termeni prin ale căror valori pot fi realizate diverse tipuri de configurații [44]:

- Configurația în formă de coloană: aceasta reprezintă forma tradițională a plutoanelor în care vehiculele sunt plasate unul în fața celuilalt (Fig. 2 a));
- Configurația în linie: vehiculele sunt așezate unul lângă celălalt. În acest caz distanța longitudinală este nulă (Fig. 2 b));
- Configurație de tip eșalon: în acest caz vehiculele sunt într-o coloană, dar fiecare este deplasat cu o anumită distanță laterală față de vehiculul precedent (Fig. 2 c));
- Configurații arbitrare: formația în formă de pană. Configurațiile arbitrare pot avea mai multe forme obținute prin combinarea a două sau mai multe din configurațiile descrise anterior (Fig. 2 d)). Acestea sunt cel mai des întâlnite în medii militare.

Cu ajutorul vehiculelor care au implementate sistemele de control automat este posibilă crearea unui pluton de vehicule care să se deplaseze în mod automat prin urmărirea unui vehicul-lider ce ar putea fi controlat automat prin comunicații cu infrastructura sau manual de către un șofer. Actualul proiect tratează deplasarea vehiculelor într-o formație de tip coloană, acțiunea fiind astfel posibilă doar pe o singură bandă de rulare dacă se consideră doar deplasarea înainte. În cazul în care acțiunile vehiculelor din pluton presupun și o schimbare a benzii de mers (o ieșire din pluton) intervine și controlul lateral.

Scopul organizării autovehiculelor în plutoane reprezintă o metodă pentru creșterea capacității drumurilor. Acest tip de organizare permite scăderea distanțelor dintre vehicule utilizând cuplaj electronic printr-o serie de dispozitive care să facă facil acest lucru. Această funcționalitate permite mai multor mașini să accelereze sau să frâneze simultan. Autovehiculele inteligente sunt capabile să se alăture unui pluton sau să-l părăsească în mod automat. O soluție de tehnologie propusă în alte lucrări de specialitate pentru realizarea acestui lucru este o infrastructură automatizată (Automated Highway System AHS). Potențialele beneficii ale AHS sunt: economia de combustibil, reducerea aglomerației, schimburi scurte în perioadele de vârf, mai puține coliziuni în trafic, și abilitatea vehiculelor de a fi conduse nesupravegheat [9].

2.2.4 Comunicații între vehicule

O variantă alternativă la sistemul AHS este implementarea unui tip de comunicații între vehiculele din cadrul unui pluton pentru a face mai eficientă deplasarea acestora. Cea de-a doua propunere reprezintă obiectul acestui proiect care se urmărește a fi dezvoltat pentru a aduce îmbunătățiri la sistemele de control deja existente pentru plutoanele de autovehicule.

Alegerea soluției de comunicații între vehicule este o variantă mai puțin costisitoare decât construirea unei întregi infrastructuri de transport în scopul îndeplinirii acestui obiectiv. Îmbunătățirile care sunt aduse la un pluton clasic de autovehicule care au sisteme ACC pentru controlul longitudinal sunt reprezentate de adăugarea unor sisteme de comunicații wireless care se numesc rețele ad-hoc de vehicule (Vehicular Ad-hoc Network VANET). Prin cuplarea celor două tehnologii de control s-a ajuns la crearea unui sistem cooperativ de control a vitezei de croazieră (Cooperative Adaptive Cruise Control CACC) care este varianta cea mai avansată a CC.

Deși deplasarea în pluton oferă o mulțime de beneficii, aceasta necesită cooperare între vehicule cu ajutorul unui protocol de management al plutoanelor. Un astfel de protocol bine dezvoltat este foarte important pentru a asigura performanțe bune ale CACC și care permit verificarea în aplicații din lumea reală. În lucrarea [45] se prezintă dezvoltarea unui protocol de organizare a plutoanelor care presupune ca vehiculele să fie capabile să execute trei manevre: unirea unor plutoane, despărțirea lor și schimbarea benzii de deplasare. Aceste trei manevre de bază pot fi folosite pentru a îndeplini diverse operațiuni ale plutonului cum ar fi intrarea unui vehicul în pluton, a liderului, și a vehiculelor următoare. Protocolul se bazează pe comunicațiile vehicul-la-vehicul (V2V) cu mesaje de tip single-hop la fel ca mesajele bazate pe evenimente pentru a coordona manevrele cu alte vehicule vecine.

Ca o parte integrală a unui sistem inteligent de transport (Intelligent Transportation System ITS) (Fig. 3), comunicația între vehicule (Inter Vehicle Communication IVC) combină următoarele tehnologii [46]:

- detectarea și perceperea mediului apropiat: prin utilizarea diferitelor tipuri de senzori (pentru condițiile vremii, starea drumului, starea vehiculului, poluare și altele) și camere, șoferul deține o anumită cantitate de informații și o vizibilitate mai bună în interiorul vehiculului propriu, făcându-l capabil să reacționeze adecvat la schimbările din mediul apropiat;
- procesarea: cu o capacitate de procesare foarte mare, vehiculele curente sunt inteligente și pot interpreta informațiile conectate cu scopul de a ajuta șoferul să ia deciziile adecvate;
- salvarea: este necesar un spațiu mare de stocare pentru a se putea salva cantități mari de informații de diverse tipuri și clase. Aceste structuri de date sunt actualizate prin evenimentele și deciziile care apar de la sistemul de comunicații;
- rutarea și comunicația: pentru schimbul de informații și răspândirea în rețeaua proprie de vehicule sau în alte rețele. Aceasta permite creșterea ariei de precauție cu ajutorul unei percepți extinse a mediului, obținându-se astfel o predicție cât mai bună a problemelor întâlnite de-a lungul deplasării.

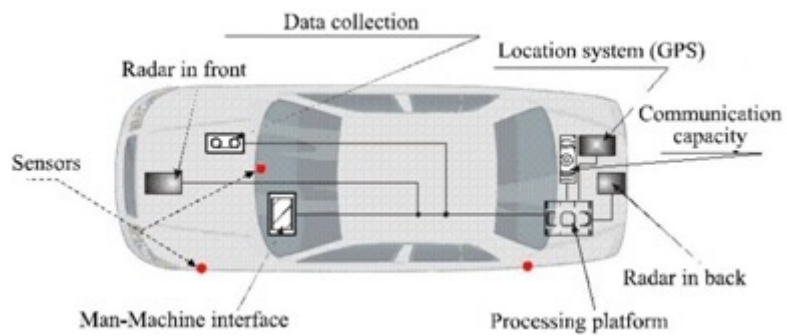


Figura 3: Vehicul inteligent cu dispozitivele de automatizare.

Etapa I

Dezvoltarea unui model realist distribuit pentru plutoanele de autovehicule

I.1 Obiectivele etapei

Obiectiv 1: Dezvoltarea unui model realist distribuit pentru plutoanele de autovehicule

- **Activitate 1.1.** Selectarea celor mai recente modele pentru plutoanele de vehicule și dezvoltarea de noi modele distribuite ce evidențiază cuplajele dintre vehicule
- **Activitate 1.2.** Modelarea întârzierilor variabile în timp din rețea folosind aproximări politopice sau considerându-le ca perturbații pentru a le exprima ca partiții poliedrale ale spațiului stărilor
- **Activitate 1.3.** Modelarea restricțiilor hardware ca partiții poliedrale ale spațiului stărilor
- **Activitate 1.4.** Integrarea optimă a acestor modele într-un model distribuit realist pentru PV
- **Activitate 1.5.** Analiza modelului complet din punctul de vedere al strategiilor de control
- **Livrabil 1:** Raport cu descrierea modelului întârzierilor și a restricțiilor.
- **Livrabil 2:** Raport cu descrierea modelului complet al vehiculelor cooperante.

I.2 (A1.1) Selectarea celor mai recente modele pentru plutoanele de vehicule și dezvoltarea de noi modele distribuite ce evidențiază cuplajele dintre vehicule

I.2.1 Stadiul actual referitor la metodele de modelare a dinamicii vehiculelor

În contextul planificării traiectoriei și controlului vehiculelor autonome, scopul unui model al vehiculului este de a oferi o relație între semnalele de intrare fizice și orientarea vehiculului. Cele mai des utilizate modele de vehicule, în funcție de complexitate, sunt clasificate în trei categorii:

- *modele de tip "mass point"*, în care vehiculul este redus la un punct în spațiu. Acesta este cel mai simplu model de vehicul, dar în același timp nu este utilizat foarte mult în aplicațiile de planificare de traiectorie [47] și de control al vehiculelor deoarece nu poate surprinde dinamica vehiculului cu acuratețea necesară.
- *modele cinematice* [48, 49] ce pot fi derivate ușor folosind relațiile geometrice care descriu un vehicul. Aceste modele pot reprezenta acceptabil mișcarea vehiculelor în situații de funcționare care nu presupun o dinamică ridicată și condiții nominale de rulare. Aceste modele sunt foarte atractive pentru a descrie dinamica vehiculului la viteze mici, deoarece, în comparație cu modelele dinamice (prezentate mai jos) nu prezintă singularități în calcularea soluției (stările vehiculului).
- *modele dinamice* [48, 49] care se bazează pe modele ale cauciucurilor pentru a descrie interacțiunea dintre vehicul și suprafața de rulare. Astfel, acest tip de model este mai complex decât cele amintite anterior, deoarece include și neliniaritățile apărute datorită interacțiunilor dintre forțele generate de cauciucuri și stările vehiculului. Aceste modele sunt cele mai utilizate în aplicații. De asemenea, în funcție de aplicații, aceste modele pot fi simplificate și liniarizate, pentru a surprinde doar dinamicile de interes. La rândul lor modelele dinamice variază în funcție de gradele de libertate considerate:
 - modele dinamice care consideră toate cele patru roți ale vehiculului;
 - modele dinamice în care cele patru roți sunt reduse la două roți modelul de tip bicicletă;
 - modele care surprind dinamica doar pe orizontală;
 - modele care surprind dinamica atât pe orizontală cât și pe verticală.

În acest raport se consideră un model dinamic de tip bicicletă al vehiculului, deoarece acest model poate surprinde cu o acuratețe acceptabilă cele mai importante dinamici ale vehiculului, surprinzând doar mișcarea pe orizontală. Acest model va fi detaliat în subcapitolele următoare.

Modelele dinamice [50] ale vehiculelor folosesc modele de cauciucuri pentru a calcula forțele generate de roțile vehiculului, care produc mișcarea. Aceste modele sunt variate și pot fi alese în funcție de aplicații. Cel mai complet model al cauciucului este așa numitul model "formula magică" sau "modelul Pacejka" [51, 52]. Acesta este un model semi-empiric, des utilizat în aplicații. În subcapitolele următoare va fi descris în detaliu și acest model.

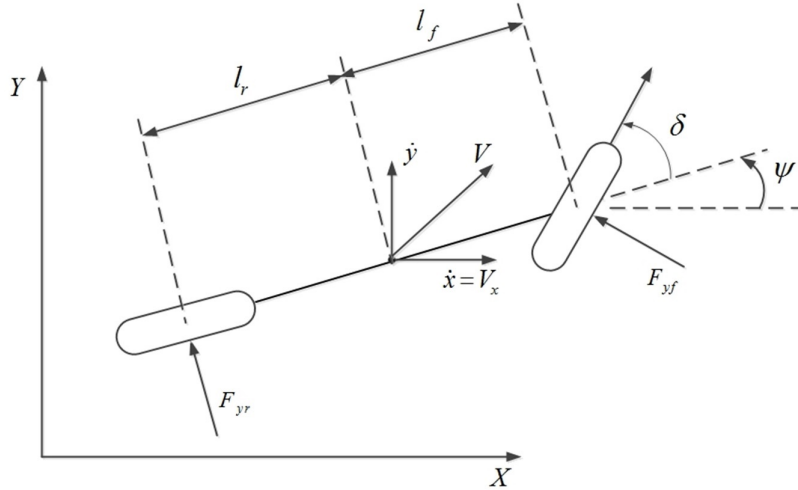


Figura I.1: Modelul biciletă al dinamicii vehiculului.

I.2.1.1 Dinamica laterală a vehiculelor

Un model foarte des utilizat pentru a descrie dinamica vehiculelor este modelul biciletă (eng. bicycle model sau single track model) [53]. În acest model, perechile de roți din față și spate ale mașinii sunt reduse la câte o singură roată, așa cum este reprezentat în Fig. I.1, unde vehiculul este considerat în reperul inerțial (OXY).

Considerând că vehiculul modelat poate vira doar cu roțile din față, în acest model δ reprezintă unghiul de virare al roții. ψ este unghiul de rotație al vehiculului, V este viteza de deplasare, F_{yf} și F_{yr} sunt forțele laterale generate de roțile din față și din spate, respectiv. l_f și l_r sunt distanțele de la centrul de greutate la centrul punții din față și din spate. V_x este viteza longitudinală a vehiculului.

Modelul are două grade de libertate, acestea fiind: poziția laterală a vehiculului y și unghiul de rotație ψ , iar unghiul de virare δ este considerat intrare în model. Poziția laterală este măsurată de-a lungul axei Y a reperului inerțial, fiind coordonată pe această axa a centrului de greutate al vehiculului. ψ este măsurat având ca referință axa X a reperului inerțial.

Aplicând legea a doua a lui Newton, putem obține ecuațiile dinamicii vehiculului pe axa Y

$$m\ddot{y} = -m\dot{\psi}V_x + 2F_{yf} + 2F_{yr}, \quad (I.1)$$

unde m este masa vehiculului. Se poate observa că în mișcarea laterală apar două accelerații - accelerația \ddot{y} datorată deplasării de-a lungul axei Y și accelerația centripetă $\dot{\psi}V_x$.

$$I_z\ddot{\psi} = 2l_fF_{yf} - 2l_rF_{yr}, \quad (I.2)$$

unde I_z este inerția vehiculului.

În continuare vor fi descrise forțele laterale care acționează asupra vehiculului, F_{yf} și F_{yr} . Aceste forțe sunt principalele influențe externe care acționează asupra vehiculului și sunt puternic neliniare, dar pot fi liniarizate considerând unghiuri de alunecare mici ale cauciucurilor roților din față și din spate. Unghiul de alunecare al cauciucului este definit ca

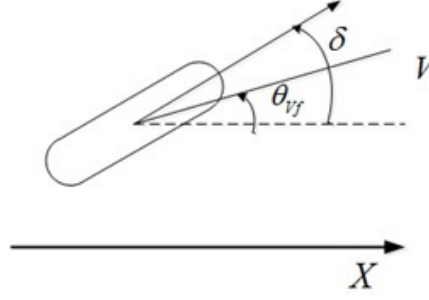


Figura I.2: Vehicul inteligent cu dispozitivele de automatizare.

unghiul dintre orientarea cauciucului și vectorul de viteză a roții. În Fig. I.2 este reprezentat unghiul de alunecare al cauciucului roții din față θ_{Vf} , unghiul de alunecare al cauciucului roții din spate având aceeași definiție.

Unghiul de alunecare pentru cauciucul roții din față este definit ca

$$\alpha_f = \delta - \theta_{Vf}. \quad (I.3)$$

Considerând un vehicul care virează doar cu roțile din față, unghiul de alunecare al cauciucului roții din spate este descris prin relația:

$$\alpha_r = -\theta_{Vr}. \quad (I.4)$$

Astfel, forțele laterale F_{yf} și F_{yr} sunt date de

$$\begin{aligned} F_{yf} &= 2C_{\alpha f}(\delta - \theta_{Vf}), \\ F_{yr} &= 2C_{\alpha r}(-\theta_{Vr}). \end{aligned} \quad (I.5)$$

Constantele de proporționalitate $C_{\alpha f}$ și $C_{\alpha r}$ din (I.5), sunt constantele de rigiditate la viraje ale cauciucurilor celor două roți.

Unghiurile θ_{Vf} și θ_{Vr} pot fi calculate folosind relațiile,

$$\begin{aligned} \tan(\theta_{Vf}) &= \frac{V_y + l_f \dot{\psi}}{V_x}, \\ \tan(\theta_{Vr}) &= \frac{V_y - l_f \dot{\psi}}{V_x}, \end{aligned} \quad (I.6)$$

care, utilizând aproximațiile pentru unghiuri mici și ținând cont că $V_y = \dot{y}$, pot fi rescrise ca

$$\begin{aligned} \theta_{Vf} &= \frac{V_y + l_f \dot{\psi}}{V_x}, \\ \theta_{Vr} &= \frac{V_y - l_f \dot{\psi}}{V_x}. \end{aligned} \quad (I.7)$$

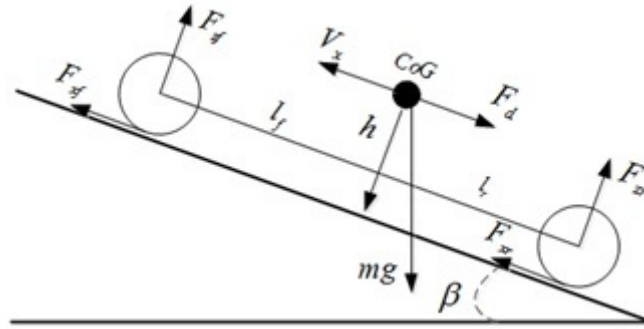


Figura I.3: Modelul longitudinal al unui vehicul.

Modelul prezentat poate fi rescris sub forma intrare-stare-ieșire astfel

$$\begin{bmatrix} \dot{y} \\ \ddot{y} \\ \dot{\psi} \\ \ddot{\psi} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{2C_{\alpha f} + 2C_{\alpha r}}{mV_x} & 0 & -\frac{2l_f C_{\alpha f} - 2l_r C_{\alpha r}}{mV_x} \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{2l_f C_{\alpha f} - 2l_r C_{\alpha r}}{I_z V_x} & 0 & -\frac{2l_f^2 C_{\alpha f} + 2l_r^2 C_{\alpha r}}{I_z V_x} \end{bmatrix} \begin{bmatrix} y \\ \dot{y} \\ \psi \\ \dot{\psi} \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{2C_{\alpha f}}{m} \\ 0 \\ \frac{2l_f C_{\alpha f}}{I_z} \end{bmatrix} \delta. \quad (I.8)$$

Acest model este un model liniar, cu parametri variabili în timp datorită vitezei longitudinale V_x .

Cu toate că acest model este în general suficient pentru studiul și proiectarea sistemelor de control convenționale ale vehiculelor, precum sistemele de avertizare a ieșirii de pe banda de rulare, sistemele active de păstrare a benzii sau sisteme de stabilizare a vehiculului, în controlul vehiculelor autonome, presupunerea că unghiul de alunecare al cauciucului are valori mici nu mai poate fi făcută. Cu această constrângere, forțele F_{yf} și F_{yr} nu mai pot fi modelate de relațiile (I.5), având o formă neliniară. Un model des folosit al forțelor cauciucurilor F_{yf} și F_{yr} este modelul Pacejka [51, 52], cunoscut și ca formula magică, care descrie forțele laterale folosind atât unghiul de alunecare, cât și forțele normale care acționează asupra roților.

I.2.1.2 Dinamica longitudinală a vehiculelor

Pentru a studia dinamica longitudinală vom considera un vehicul rulând pe un drum înclinat, așa cum este prezentat în Fig. I.3. Forțele externe care acționează asupra corpului vehiculului în mișcare sunt forța aerodinamică, forța gravitațională, forța de frecare a căii de rulare și forțele longitudinale ale cauciucurilor. Toate aceste forțe vor fi prezentate în detaliu în cele ce urmează.

În Fig. I.3 CoG reprezintă centrul de greutate al vehiculului, F_{xf} și F_{xr} reprezintă forțele longitudinale ale cauciucurilor, F_{zf} și F_{zr} sunt forțele normale ale axelor din față și din spate, V_x este viteza longitudinală a vehiculului și F_d este forța rezistentă aerodinamică. h reprezintă înălțimea centrului de greutate, l_f și l_r au aceeași semnificație ca și în modelul dinamicii laterale a vehiculului fiind distanțele de la centrul de greutate la axa față și spate, m este masa vehiculului și g este accelerația gravitațională. Unghiul β reprezintă panta

căii de rulare a vehiculului și este definit pozitiv atunci când vehiculul se deplasează spre stânga, în Fig. I.3.

Echilibrul de forțe care acționează de-a lungul axei longitudinale este

$$m\ddot{x} = 2F_{xf} + 2F_{xr} - F_d - mg\sin(\beta). \quad (\text{I.9})$$

Forța rezistentă aerodinamică F_d poate fi reprezentată ca

$$F_d = \frac{1}{2}\rho C_d A_f (V_x + V_w)^2, \quad (\text{I.10})$$

unde este densitatea aerului și poate fi considerată $= 1.225 \text{ kg/m}^3$, C_d este coeficientul de rezistență aerodinamică, A_f este aria frontală a vehiculului și V_w este viteza vântului.

Forțele longitudinale ale cauciucurilor F_{xf} și F_{xr} sunt forțe de frecare ale suprafeței de rulare, care acționează asupra cauciucurilor. Ele depind de procentul de alunecare, forțele normale care acționează asupra roții, și de coeficientul de frecare cu drumul.

Diferența dintre viteza longitudinală a vehiculului măsurată la axa roții (V_x) și viteza longitudinală echivalentă a roții $r_{eff}\omega_w$ se numește alunecare longitudinală. Adică alunecarea longitudinală este: $r_{eff}\omega_w - V_x$. În acest sens, procentul de alunecare longitudinală este definit ca:

$$\begin{aligned} \sigma &= \frac{r_{eff}\omega_w - V_x}{V_x} - \text{în timpul frânării,} \\ \sigma &= \frac{r_{eff}\omega_w - V_x}{r_{eff}\omega_w} - \text{în timpul accelerării.} \end{aligned} \quad (\text{I.11})$$

Dacă coeficientul de frecare dintre cauciuc și suprafața de rulare este considerat 1, și forțele normale F_{zf} și F_{zr} sunt constante, iar procentul de alunecare al cauciucului este considerat mic, în general sub 0.1 pe suprafețe de rulare uscate, forțele F_{xf} și F_{xr} se pot obține ca fiind proporționale cu procentele de alunecare ale cauciucurilor roților din față și din spate:

$$\begin{aligned} F_{xf} &= C_{\sigma f} \sigma_{xf}, \\ F_{xr} &= C_{\sigma r} \sigma_{xr}, \end{aligned} \quad (\text{I.12})$$

unde $C_{\sigma f}$ și $C_{\sigma r}$ sunt coeficienți de rigiditate longitudinală a cauciucurilor din față și din spate respectiv.

Dacă procentele de alunecare ale cauciucurilor nu sunt mici, atunci acest model liniar al forțelor longitudinale nu mai poate fi folosit. Ca și în cazul forțelor laterale prezentate în secțiunea anterioară, un model neliniar al forțelor longitudinale poate fi obținut folosind modelul de cauciuc al lui Pacejka [51, 52].

Forțele normale care acționează asupra roților se pot calcula astfel:

$$\begin{aligned} F_{zf} &= \frac{-F_d h - m\ddot{x}h - mgh \sin \beta + mgl_r \cos \beta}{2(l_f + l_r)}, \\ F_{zr} &= \frac{F_d h + m\ddot{x}h + mgh \sin \beta + mgl_r \cos \beta}{2(l_f + l_r)}. \end{aligned} \quad (\text{I.13})$$

Deci, când vehiculul accelerează, forța normală care acționează pe roata din față scade, și forța normală care acționează pe roata din spate crește, atunci când β este pozitiv.

Folosind relațiile de la (I.9) până la (I.13), se obține modelul matematic al dinamicii longitudinale al vehiculului, având ca ieșire poziția x pe axa longitudinală a reperului cartezian considerat, iar ca intrare viteza unghiulară a roților. Astfel, forța longitudinală a roții de conducere, în funcție de configurația transmisiei, devine principala forță care generează mișcarea vehiculului. Viteza unghiulară a roții de conducere este generată de lanțul de transmisie a vehiculului, astfel pentru a obține un model integral al unui vehicul, trebuie considerat și lanțul de transmisie.

I.2.1.3 Modelarea forțelor cauciucurilor cu modelul Pacejka

Forțele și cuplurile apărute datorită suprafeței de rulare a vehiculului acționează asupra cauciucurilor și influențează foarte mult dinamica acestuia. Spre deosebire de o roată rigidă și nedeformabilă, cauciucul nu face contact cu suprafața de rulare doar într-un singur punct. Cauciucul se deformează datorită forțelor verticale care acționează asupra lui și face contact cu suprafața de rulare pe o arie, diferită de zero, numită arie de contact.

Forțele generate de cauciucuri, adică forțele laterale F_{yf} , F_{yr} , și forțele longitudinale F_{xf} , F_{xr} sunt neliniare, aceste neliniarități având cele mai importante influențe în manevrabilitatea și stabilitatea vehiculului. Într-un vehicul autonom, sistemul de control trebuie să poată ține cont de aceste influențe cu o precizie cât mai ridicată, pentru a asigura confortul pasagerilor și siguranța tuturor participanților la trafic (Carvalho et al., 2015). Astfel, pentru aceste sisteme de control, modelele forțelor generate de cauciucuri (4.5), (4.6), (4.16) și (4.17) devin insuficiente, atât din punct de vedere al studiului dinamicii vehiculului prin simulare, cât și din punctul de vedere al sistemului de control al vehiculului autonom care trebuie să fie capabil să prezică cât mai precis aceste forțe.

Modelul Formula Magică [51, 52], oferă o metodă de a calcula forțele F_{yf} , F_{yr} , F_{xf} și F_{xr} pentru un orizont larg de condiții de operare, inclusiv pentru unghiuri de alunecare și procente de alunecare mari ale cauciucurilor.

În cazul simplificat când doar forțele laterale F_{yf} , F_{yr} sau forțele longitudinale F_{xf} , F_{xr} trebuie să fie calculate, poate fi folosită formula:

$$Y(X) = y(x) + s_v, \quad (\text{I.14})$$

unde

$$\begin{aligned} y(x) &= D \sin \left\{ C \tan^{-1} \left[B(1 - E)x + E \tan^{-1}(Bx) \right] \right\}, \\ x &= X - s_h \end{aligned} \quad (\text{I.15})$$

cu $Y(X)$ fiind o funcție care descrie forța modelată, care poate fi forța longitudinală F_x sau forța laterală F_y a cauciucului, iar X fiind procentul de alunecare sau unghiul de alunecare.

Parametrii acestui model sunt: B factorul de rigiditate, C factorul de formă, D valoarea maximă, E factorul de curbură, s_v deplasamentul pe verticală și s_h deplasamentul pe orizontală.

Acesta este un model semi-empiric, care poate produce caracteristici care aproximează cu precizie forțele F_x și F_y pe baza valorilor variabilelor σ și α .

Pentru unghiuri de alunecare sau procente de alunecare mici, modelul descris de (I.14)-(I.15) devine

$$Y = (BCD)X, \quad (I.16)$$

unde produsul BCD reprezintă coeficientul de rigiditate laterală (C_α) sau coeficientul de rigiditate longitudinală (C_σ).

Considerând forțele normale constante, parametrii acestui model pot fi exprimați ca funcții de această forță și de unghiul de cambrare al roții, iar forțele F_x și F_y vor avea în final forma:

Pentru forța laterală:

$$F_y = D \sin \left\{ C \tan^{-1} \left[B(1 - E)(\alpha + s_h) + E \tan^{-1} (B(\alpha + s_h)) \right] \right\} + s_v, \quad (I.17)$$

cu

$$\begin{aligned} D &= a_1 F_z^2 + a_2 F_z, \\ B &= \frac{a_3 \sin [a_4 \tan^{-1} (a_5 F_z)]}{CD}, \\ E &= a_6 F_z^2 + a_7 F_z + a_8, \\ C &= 1.3, \\ s_h &= a_9 \gamma, \\ s_v &= (a_{10} F_z^2 + a_{11} F_z) \gamma, \end{aligned} \quad (I.18)$$

unde parametrii $a_i, i = 1, \dots, 11$ sunt constanți și trebuie determinați experimental, iar γ reprezintă unghiul de cambrare.

Pentru forța longitudinală:

$$F_x = D \sin \left\{ C \tan^{-1} \left[B(1 - E) \sigma + E \tan^{-1} (B\sigma) \right] \right\}, \quad (I.19)$$

cu

$$\begin{aligned} B &= \frac{a_3 F_z^2 + a_4 F_z}{C D e^{a_5 F_z}}, \\ C &= 1.65, \end{aligned} \quad (I.20)$$

iar parametrii D și E sunt dați în (I.18). Ca și la forța laterală, coeficienții $a_i, i = 1, \dots, 11$ sunt constanți.

Modelarea forțelor cauciucurilor pe baza modelului formulei magice este mai completă decât modelarea liniară. Un dezavantaj al acestei formulări este faptul că modelul presupune forțele normale care acționează asupra roților constante, ceea ce nu este întotdeauna îndeplinit atunci când vehiculul se deplasează, așa cum reiese din (I.13).

Folosind acest model al cauciucului în modelarea completă a dinamicii vehiculului, se obține unui model neliniar pentru autovehicul.

I.2.1.4 Modelarea dinamicii complete a vehiculului

În subcapitolele anterioare au fost prezentate dinamica laterală și dinamica longitudinală a vehiculului decuplate una față de cealaltă, ceea ce în realitate nu este valabil. În acest subcapitol va fi studiat modelul dinamic complet al unui autovehicul, care include atât dinamica

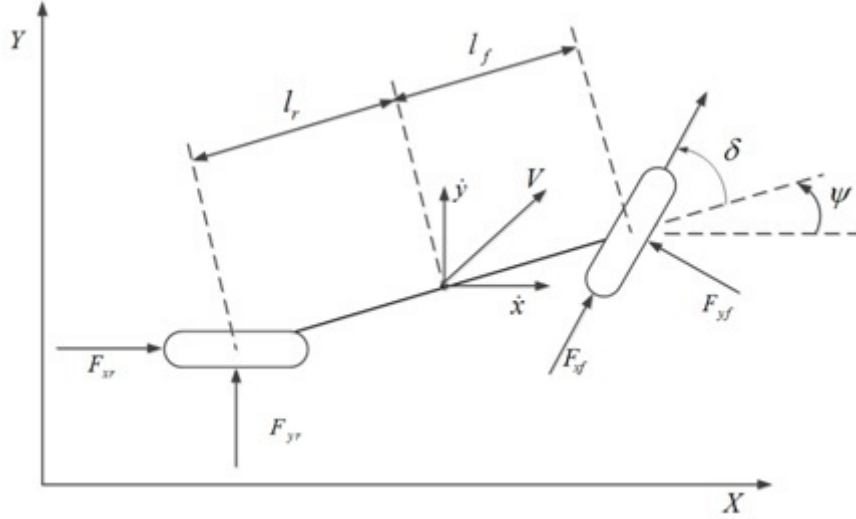


Figura I.4: Modelul complet al unui vehicul.

laterală, cât și cea longitudinală. Folosind legea a doua lui Newton, și câteva transformări geometrice, modelul complet al unui autovehicul poate fi construit, considerând dinamica laterală, descrisă de ecuațiile (I.1) și (I.2) și dinamica laterală descrisă de ecuația (I.9), astfel obținând un model de tip bicicletă extins de forma:

$$\begin{aligned}
 m\ddot{x} &= m\dot{y}\dot{\psi} + 2F_{xf} + 2F_{xr} - F_d - mg\sin(\beta), \\
 m\ddot{y} &= -m\dot{\psi}\dot{x} + 2F_{yf} + 2F_{yr}, \\
 I_z\ddot{\psi} &= 2l_f F_{yf} - 2l_r F_{yr}.
 \end{aligned} \tag{I.21}$$

Considerând același reper inerțial, modelul de tip bicicletă (I.21) este prezentat în Fig. I.4.

Modelul dinamic (I.21) poate fi rescris într-un formalism intrare-stare-ieșire neliniar de forma

$$\dot{x}(t) = f(x(t), u(t)), \tag{I.22}$$

cu condiția inițială:

$$x(0) = x_0, \tag{I.23}$$

unde $x \in \mathbb{R}^n$ reprezintă vectorul stărilor, $u \in \mathbb{R}^m$ reprezintă vectorul intrărilor și $f : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$ este o funcție neliniară care descrie dinamica sistemului.

Ieșirea modelului neliniar intrare-stare-ieșire poate fi generic descrisă de:

$$y(t) = h(x(t), u(t)), \tag{I.24}$$

unde $y \in \mathbb{R}^r$ este vectorul ieșirilor, iar $h : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^r$ poate fi liniară sau neliniară.

Modelul (I.21), poate fi adus în forma (I.22)-(I.24) considerând ca stări ale sistemului poziția longitudinală $x(t)$, viteza longitudinală $\dot{x}(t)$, poziția laterală $y(t)$, viteza laterală

$\dot{y}(t)$, unghiul de rotație $\psi(t)$ și viteza de rotație $\dot{\psi}(t)$ astfel:

$$\begin{aligned}
\dot{x}_1 &= x_2 \\
\dot{x}_2 &= x_4 x_6 + \frac{2}{m} F_{xf} + \frac{2}{m} F_{xr} - \frac{2}{m} F_d - g \sin(\beta) \\
\dot{x}_3 &= x_4 \\
\dot{x}_4 &= -x_2 x_6 + \frac{2}{m} F_{yf} + \frac{2}{m} F_{yr} \\
\dot{x}_5 &= x_6 \\
\dot{x}_6 &= \frac{2}{I} (l_f F_{yf} - l_r F_{yr}),
\end{aligned} \tag{I.25}$$

unde $x = [x_1 x_2 x_3 x_4 x_5 x_6]^T = [x \dot{x} y \dot{y} \psi \dot{\psi}]^T$.

Pentru calcularea forțelor produse de cauciucuri, va fi folosit modelul Pacejka (prezentat anterior), extins. În continuare va fi prezentat modelul extins al cauciucurilor.

Modelul Pacejka [51, 52] prezentat în subsecțiunea anterioară descrie forțele laterale și longitudinale ale cauciucului în condiții staționare, adică, pentru forțele laterale presupune ca alunecarea longitudinală este nulă, iar la forțele longitudinale se consideră ca unghiul de alunecare este nul. Astfel, cele două forțe fiind complet decuplate.

Un alt dezavantaj al modelului Pacejka prezentat în subsecțiunea anterioară este faptul că nu ia în considerare variațiile forțelor normale, în calculul parametrilor modelului, astfel într-o interpretare fizică, presupunând că anvelopele vehiculului nu își modifică forma în funcție de variația forțelor normale. Trebuie menționat că aceste forțe normale variază în funcție de încărcarea vehiculului, suprafața de rulare și de manevrele realizate de vehicul - accelerare, frânare și virare, deoarece în aceste manevre, centrul de greutate al vehiculului se va deplasa de la poziția nominală.

Efectul unei forțe normale diferite față de forța normală nominală F_{z0} (forțele normale generate doar de greutatea vehiculului rulând pe un carosabil perfect plan), poate fi aproximat folosind un factor de scalare $\lambda_{F_{z0}}$:

$$F'_{z0} = \lambda_{F_{z0}} F_{z0}, \tag{I.26}$$

Mai departe definim variația forței normale:

$$df = \frac{F_z - F'_{z0}}{F'_{z0}}, \tag{I.27}$$

unde F_z este dată de (I.13).

Unghiul de alunecare al roții descris de (I.3), respectiv (I.4), nu produce cea mai realistă comportare a cauciucului în cazul în care devine foarte mare sau în cazul rotirii inverse a roții. Din acest motiv, mai potrivită este folosirea alunecării laterale a cauciucului descrisă de

$$\alpha^* = \tan \alpha \operatorname{sgn} V_x = -\frac{V_y}{|V_x|}, \tag{I.28}$$

Alunecarea longitudinală este dată de (I.11).

În continuare sunt descrise ecuațiile modelului Pacejka extins [51, 52].

Forța longitudinală (considerând alunecarea laterală și longitudinală)

$$\begin{aligned}
F_x &= G_{x\alpha} F_{x0} \\
G_{x\alpha} &= \cos [C_{x\alpha} \tan^{-1} \{B_{x\alpha} \alpha_s - E_{x\alpha} (B_{x\alpha} \alpha_s - \tan^{-1}(B_{x\alpha} \alpha_s))\}] / G_{x0} \\
G_{x\alpha 0} &= \cos [C_{x\alpha} \tan^{-1} \{B_{x\alpha} S_{Hx\alpha} - E_{x\alpha} (B_{x\alpha} S_{Hx\alpha} - \tan^{-1}(B_{x\alpha} S_{Hx\alpha}))\}] \\
\alpha_s &= \alpha^* + S_{Hx\alpha} \\
B_{x\alpha} &= r_{Bx1} \cos (\tan^{-1} (r_{Bx2} \sigma)) \\
C_{x\alpha} &= r_{Cx1} \\
E_{x\alpha} &= r_{Ex1} + r_{Ex2} df \\
S_{Hx\alpha} &= r_{Hx1}
\end{aligned} \tag{I.29}$$

Forța laterală (considerând alunecarea laterală și longitudinală)

$$\begin{aligned}
F_y &= G_{y\sigma} F_{y0} + S_{Vy\sigma} \\
G_{y\sigma} &= \cos [C_{y\sigma} \tan^{-1} \{B_{y\sigma} \alpha_s - E_{y\sigma} (B_{y\sigma} \alpha_s - \tan^{-1}(B_{y\sigma} \alpha_s))\}] / G_{y\sigma 0} \\
G_{y\sigma 0} &= \cos [C_{y\sigma} \tan^{-1} \{B_{y\sigma} S_{Hy\sigma} - E_{y\sigma} (B_{y\sigma} S_{Hy\sigma} - \tan^{-1}(B_{y\sigma} S_{Hy\sigma}))\}] \\
\sigma_s &= \sigma + S_{Hy\sigma} \\
B_{y\sigma} &= r_{By1} \cos (\tan^{-1} (r_{By2} (\alpha^* - r_{By2}))) \\
C_{y\sigma} &= r_{Cy1} \\
E_{y\sigma} &= r_{Ey1} + r_{Ey2} df \\
S_{Hy\sigma} &= r_{Hy1} + r_{Hy2} df \\
S_{Vy\sigma} &= D_{Vy\sigma} \sin (r_{Vy5} \tan^{-1} (r_{Vy6} \sigma)) \\
D_{Vy\sigma} &= \mu_y F_z (r_{Vy1} + r_{Vy2} df) \cos (\tan^{-1} (r_{Vy3} \alpha^*)),
\end{aligned} \tag{I.30}$$

În modelele prezentate, F_{x0} și F_{y0} sunt date de (I.17), respectiv (I.15).

Aceste modele pot fi în continuare extinse, astfel încât să includă influența unghiului de cambrare a roții și presiunea roților. De asemenea, modelul oferă ecuații și pentru simularea cuplului de aliniere, care nu va fi discutat în acest raport.

Modelul (I.22) poate fi utilizat în dezvoltarea unor algoritmi de control predictiv nelinier [54].

I.2.1.5 Implementarea modelului în Matlab/Simulink și rezultate de simulare

În această subsecțiune va fi prezentată implementarea în mediul Matlab/SIMULINK a unui model de vehicul. Implementarea este făcută pe baza aspectelor teoretice prezentate în subcapitolele anterioare. Modelarea este făcută pe baza descrierii dinamice (I.21) și este prezentat în Fig. I.5.

În acest model a fost făcută o simplificare, considerând că atât roțile din față, cât și cele din spate produc cuplul de tracțiune, astfel eliminând nevoia de modelare dinamică a roților. Această modelare rămâne subiect pentru contribuțiile viitoare.

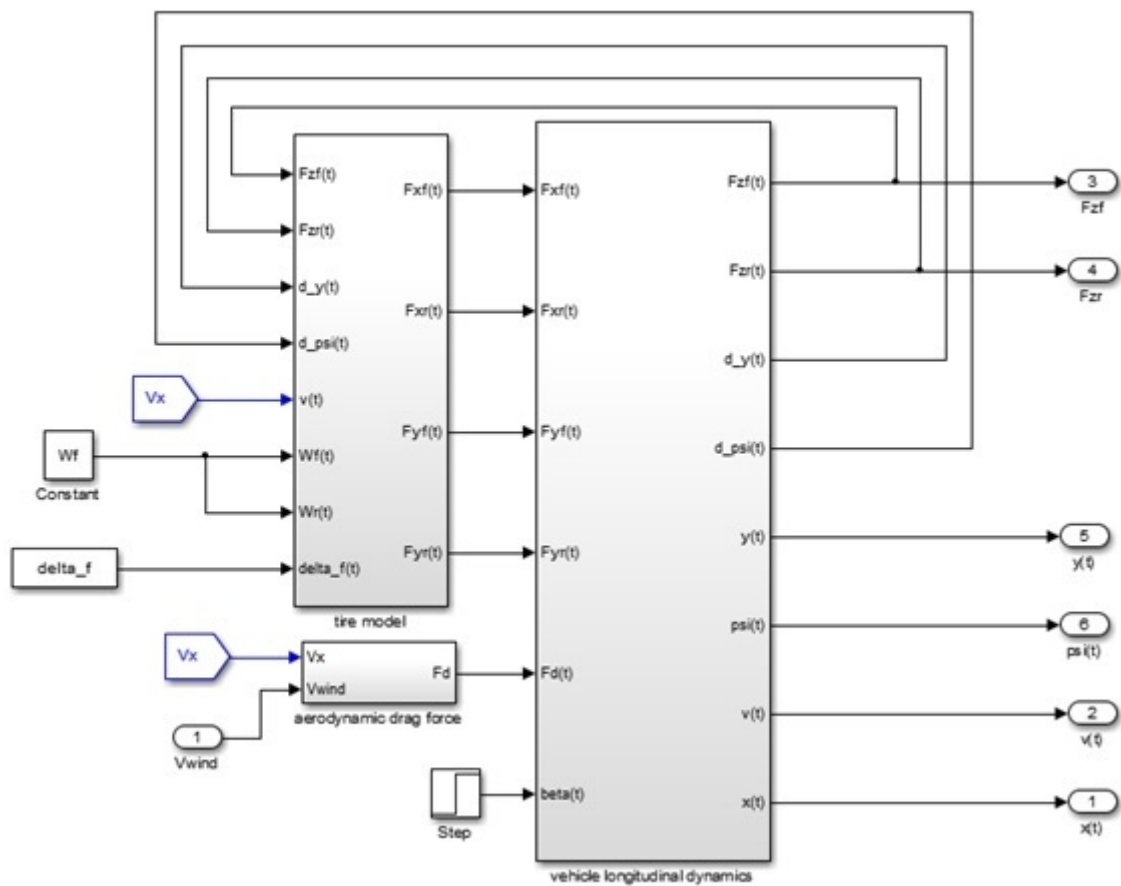


Figura I.5: Modelul dinamic al vehiculului.

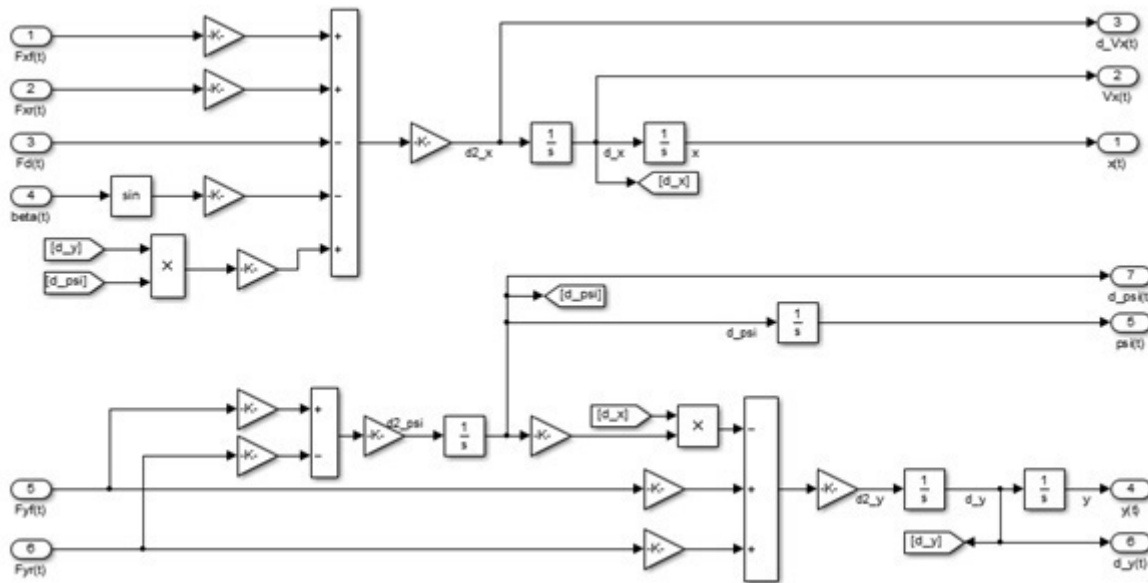


Figura I.6: Modelul de tip bicicletă al dinamicii vehiculului.

Modelul de tip bicicletă al vehiculului este reprezentat în Fig. I.6, iar modelele forțelor generate de cauciucuri, folosind modelul extins al lui Pacejka sunt ilustrate în Fig. I.7 și I.8.

În I.9 este simulată o manevră de schimbare a benzii și de revenire a unui vehiculului, cu o viteză constantă $V = 22$ km/h, arătând astfel comportamentul vehiculului în manevre de depășire sau evitare de obstacole. În această figură se poate observa traiectoria vehiculului și unghiul de virare a roții plasate pe axa din față. Lățimea benzii este considerată 3.5 m, pe un carosabil cu două sensuri având o singură bandă pe sens, deci având lățimea totală de 7 m. Poziția inițială a vehiculului este considerată fiind în centrul benzii din dreapta, deci o schimbare completă a benzii va fi obținută la poziția 3.5 m pe axa poziției laterale. Pentru a schimba banda, unghiul roții din față este crescut până când acesta ajunge la valoarea de 0.6 rad, începând din momentul $t = 5$ s. După cum se poate observa în Fig. I.9, traiectoria vehiculului începe să crească în direcția axei laterale. Când unghiul de virare este scăzut la 0 rad, poziția laterală a vehiculului rămâne constantă, vehiculul rămânând pe banda din stânga. Reîntoarcerea pe banda din dreapta se execută făcând aceeași manevră de virare în sensul opus.

În Fig. I.10 se poate observa evoluția în timp a vitezei și poziției laterale ale vehiculului. Se pot observa mici variații ale vitezei în momentele în care vehiculul virează, aceste variații fiind explicabile prin faptul că vectorul vitezei laterale este nenul în acele momente, viteza totală a vehiculului fiind suma vectorilor vitezelor pe cele două axe (Fig. I.1). De asemenea, analizând Fig. I.9 și I.10 se poate observa că poziția longitudinală a vehiculului nu este perfect constantă în momentele când unghiul de virare este nul. Această variație este datorată imperfecțiunilor cauciucurilor vehiculului, care sunt surprinse în modelul Pacejka propus.

În Fig. I.11 sunt prezentate evoluțiile în timp ale unghiului de girație și a vitezei laterale

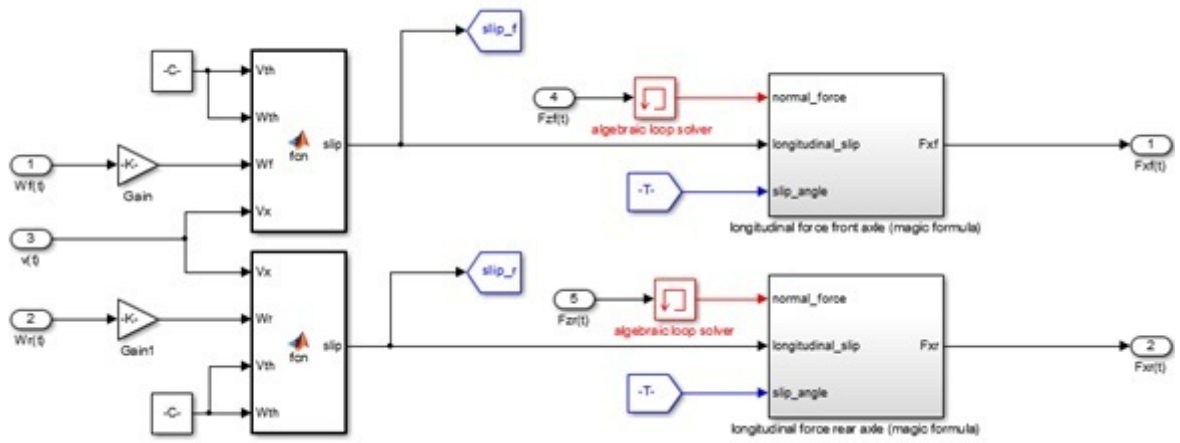


Figura I.7: Modelul forțelor longitudinale ale cauciucurilor.

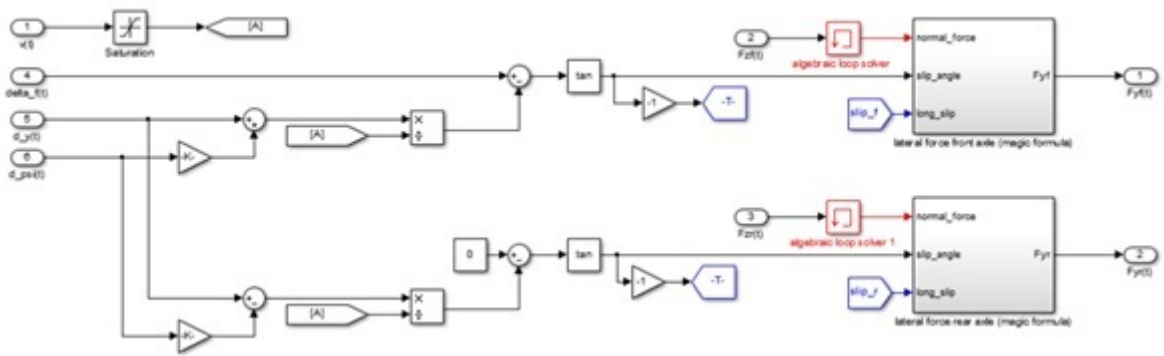


Figura I.8: Modelul forțelor laterale ale cauciucurilor.

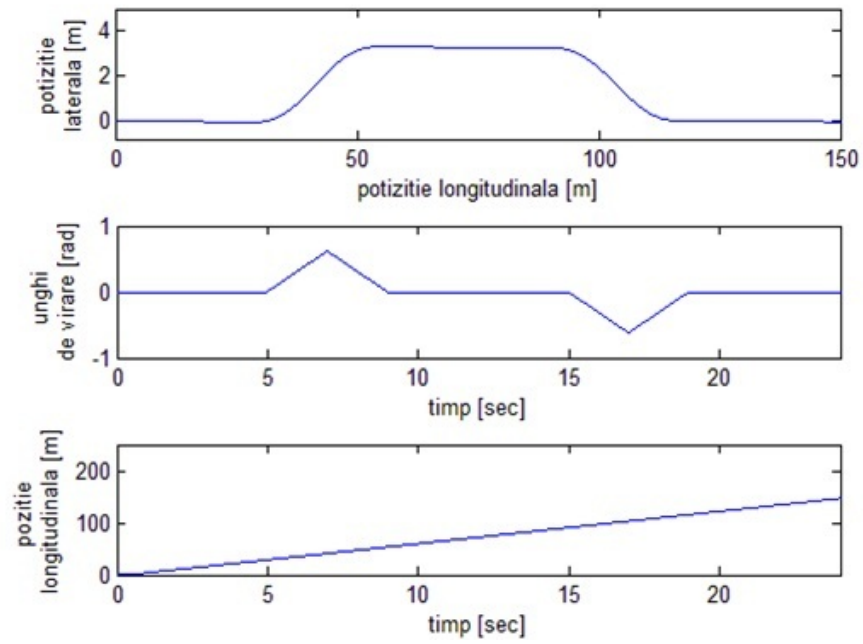


Figura I.9: Manevra de schimbare și revenire a benzii.

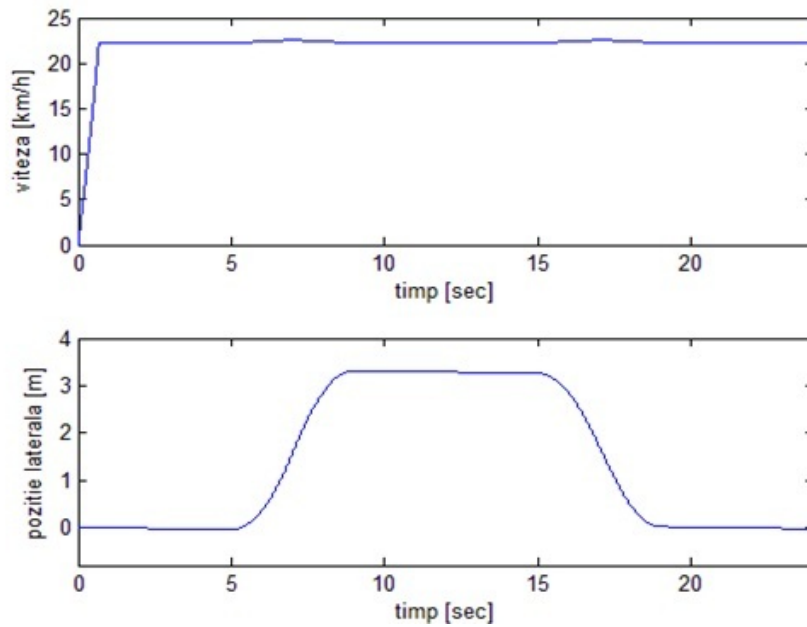


Figura I.10: Influența manevrelor de virare asupra vitezei vehiculului.

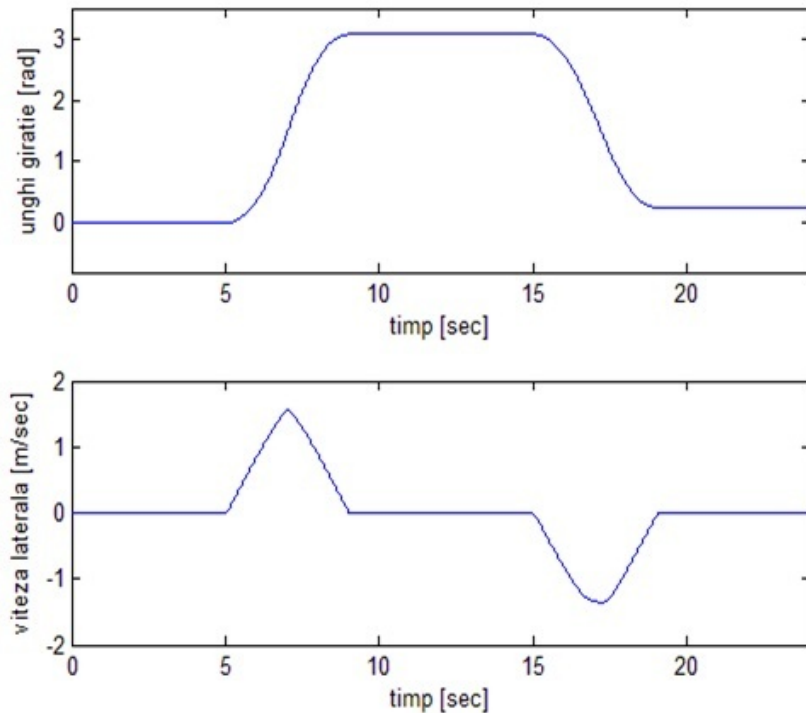


Figura I.11: Evoluțiile în timp ale unghiului de rotație și vitezei laterale.

în această manevră de schimbare dublă a benzii.

I.2.2 Modelarea lanțului de transmisie a puterii la autovehicule

Pentru dezvoltarea unui regulator, este necesară obținerea unui model precis al lanțului de transmisie a puterii, pentru a prezice răspunsul vehiculului la un anumit cuplu primit la intrare. Modelul poate fi apoi folosit pentru proiectarea sistemului de control și simularea performanțelor.

I.2.2.1 Model afin

Un model cu trei inerții, care este detaliat și în literatură [55, 56], a fost obținut folosind legile fizice ale mișcării. Prima inerție corespunde motorului, a doua inerție include inerția cutiei de viteze și a diferențialului, iar cea de-a treia inerție este corespunzătoare masei autovehiculului roților, cum se poate vedea în Fig. I.12.

Ecuțiile fundamentale ce descriu dinamica lanțului de transmisie a puterii sunt derivate folosind a doua lege de mișcare a lui Newton. Ecuația de mișcare pentru prima masă rotațională este dată de:

$$J_e \dot{\omega}_e = T_e - (k_c(\theta_e - \theta_t i_t) + d_c(\omega_e - \omega_t i_t)) - d_e \omega_e, \quad (I.31)$$

unde prima inerție este reprezentată de inerția motorului.

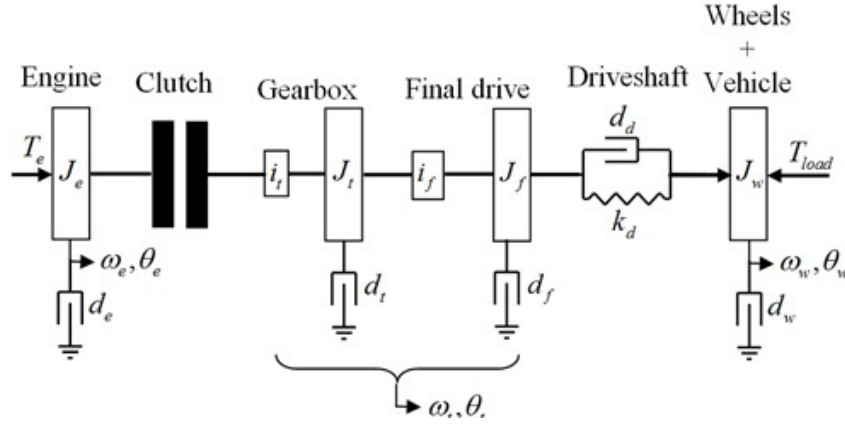


Figura I.12: Reprezentarea schematică a lanțului de transmisie a puterii.

Ecuția de mișcare pentru cea de-a doua masă rotațională poate fi scrisă sub forma:

$$(J_t + J_f/i_f^2)\dot{\omega}_t = i_t(k_c(\theta_e - \theta_t i_t) + d_c(\omega_e - \omega_t i_t)) - (d_t + d_f/i_f^2)\omega_t - \frac{1}{i_f}(k_d(\theta_t/i_f - \theta_w) + d_d(\omega_t/i_f - \omega_w)) \quad (I.32)$$

unde $J_2 = J_t + J_f/i_f^2$ reprezintă cea de-a doua inerție, cu coeficientul de amortizare $d_2 = d_t + d_f/i_f^2$.

Ultima ecuație de mișcare, cea corespunzătoare roților și masei vehiculului, poate fi scrisă sub forma:

$$(J_w + m_{COG}r_{stat}^2)\dot{\omega}_w = k_d(\theta_t/i_f - \theta_w) + d_d(\omega_t/i_f - \omega_w) - d_w\omega_w - T_{load} \quad (I.33)$$

unde $J_3 = J_w + m_{COG}r_{stat}^2$ reprezintă inerția roților și a vehiculului.

Cuplul de sarcină T_{load} este modelat ca fiind suma cuplurilor care acționează asupra autovehiculului:

$$\begin{aligned} T_{load} &= T_{roll} + T_{angle} + T_{airdrag} \\ T_{roll} &= c_{r1}m_{COG}g \cos(\chi_{road})r_{stat} \\ T_{angle} &= m_{COG}g \sin(\chi_{road})r_{stat} \\ T_{airdrag} &= 0.5\rho_{air}A_f c_d v_v^2 r_{stat} \end{aligned} \quad (I.34)$$

Pentru a reduce complexitatea modelului, influența dinamică datorată cauciucurilor poate fi neglijată în ecuația care descrie dinamica roților. În faza de modelare doar termenii dați de cuplul de rulare și cel datorat frecării cu aerul sunt luați în considerare. De asemenea, tot pentru a reduce complexitatea modelului, este folosită o aproximare liniară a cuplului datorat frecării cu aerul:

$$T_{airdrag} = c_{r2}\omega_w \quad (I.35)$$

unde c_{r2} este o constantă.

Având ca stări unghiul de torsiune dintre motor și transmisie, unghiul de torsiune dintre

transmisie și roți, precum și vitezele unghiulare ale motorului, transmisie și roților

$$\begin{aligned} x_1 &= \theta_e - \theta_t i_t \\ x_2 &= \theta_t / i_f - \theta_w \\ x_3 &= \omega_e \\ x_4 &= \omega_t \\ x_5 &= \omega_w \end{aligned}, \quad (I.36)$$

modelul intrare-stare-ieșire liniar de forma

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u} + \mathbf{f} \\ \mathbf{y} &= \mathbf{C}\mathbf{x} + \mathbf{D}\mathbf{u} \end{aligned}, \quad (I.37)$$

are următoarele matrici:

$$\mathbf{A} = \begin{bmatrix} 0 & 0 & 1 & -i_t & 0 \\ 0 & 0 & 0 & \frac{1}{i_f} & -1 \\ -\frac{k_c}{J_1} & 0 & -\frac{d_c+d_e}{J_1} & \frac{d_c i_t}{J_1} & 0 \\ \frac{k_c i_t}{J_2} & \frac{k_d}{i_f J_2} & \frac{d_c i_t}{J_2} & -\left(d_c i_t^2 + d_2 + \frac{d_d}{i_f^2}\right) & \frac{d_d}{i_f J_2} \\ 0 & \frac{k_d}{J_3} & 0 & \frac{d_d}{i_f J_3} & -\frac{(d_w + d_d + c_{r2})}{J_3} \end{bmatrix}, \quad (I.38)$$

$$\mathbf{B} = \begin{bmatrix} 0 \\ 0 \\ \frac{1}{J_1} \\ 0 \\ 0 \end{bmatrix}, \mathbf{f}_c = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ -\frac{T_{roll}}{J_3} \end{bmatrix}, \mathbf{C} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 1 \end{bmatrix}. \quad (I.39)$$

Intrarea sistemului este reprezentată de cuplul motor $u = T_e$, iar ieșirile sistemului sunt reprezentate de variabilele de stare ale acestuia.

1.2.2.2 Model afin pe porțiuni

Pornind de la modelul intrare-stare-ieșire afin (I.37)-(I.39), în modelul sistemului sunt incluse și modurile de funcționare ale ambreiajului, obținându-se în acest fel un model intrare-stare-ieșire afin pe porțiuni [57]. Ecuațiile ce descriu dinamica lanțului de transmisie a puterii sunt aceleași ca și în cazul modelului afin, dar diferite valori pentru coeficientul de elasticitate și pentru cel de amortizare sunt folosite, valori corespunzătoare modului de lucru al ambreiajului.

Analizând ambreiajul, se poate observa că există mai multe arcuri cu valori diferite ale coeficientului de elasticitate, conectate în serie, ca în Fig. I.13. Fig. I.13 a) ilustrează caracteristica de elasticitate a ambreiajului, iar structura arcurilor ambreiajului este prezentată în Fig. I.13 b). Motivul acestui aranjament este izolarea vibrațiilor care apar în cazul cuplării rapide a ambreiajului. Sunt luate în considerare două moduri ale ambreiajului: deschis și închis, modul închis având trei faze distincte, corespunzătoare unor arcuri cu coeficient de elasticitate diferit. În modul deschis ($k_{c1} = 0$), nu există nici o conexiune între motor și

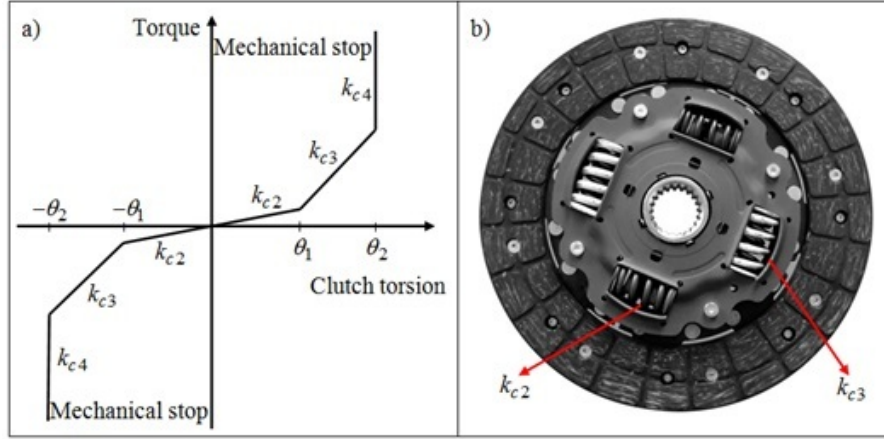


Figura I.13: Caracteristica de elasticitate a ambreiajului.

transmisie, ca urmare cuplul motor nu este transmis. În modul închis, plăcile ambreiajului se apropie, transmitând cuplul motor către roți. În prima fază a modului închis sunt compresate arcurile cu coeficient de elasticitate mic (k_{c2}). În cea de-a doua fază, arcurile cu coeficient de elasticitate mic sunt complet compresate și începe compresia arcurilor cu coeficient de elasticitate mare (k_{c3}). În ultima fază a modului închis, atât arcurile cu coeficient de elasticitate mic cât și arcurile cu coeficient de elasticitate mare sunt complet compresate, și se atinge un stop mecanic (k_{c4}).

Având aceleași stări, aceeași intrare și aceeași ieșiri ca în cazul modelului liniar, modelul intrare-stare-ieșire liniar pe porțiuni este definit de ecuația:

$$\dot{x}(t) = A_{ci}x(t) + b_c u(t) + f_c \quad \text{if } x(t) \in \Omega_i, \quad (\text{I.40})$$

unde $x := (x_1, \dots, x_5)^\top \in \mathbb{R}^5$ și $i \in I := \mathbb{Z}_{[1,4]}$. Aici i definește modul activ la momentul $t \in \mathbb{R}_+$, $A_{ci} \in \mathbb{R}^{5 \times 5}$, $b_c \in \mathbb{R}^{5 \times 1}$ sunt matricile sistemului iar $f_c \in \mathbb{R}^{5 \times 1}$ este termenul afin. Colecția de seturi $\{\Omega_i \mid i \in I\}$ definește o partiție a $X \subseteq \mathbb{R}^5$ astfel încât $\Omega_i = X$ și $\text{int}(\Omega_i) \neq \emptyset$ pentru toți $i \in I$. Regiunile sunt definite după cum urmează:

$$\left\{ \begin{array}{ll} \Omega_1 := \{x \in \mathbb{R}^5 \mid x_3 \leq \omega_e^{\text{closing}}\}, & \text{—deschis} \\ \Omega_2 := \{x \in \mathbb{R}^5 \mid x_3 > \omega_e^{\text{closing}} \mid x_1 \leq \theta_1\}, & \text{—închisI} \\ \Omega_3 := \{x \in \mathbb{R}^5 \mid x_3 > \omega_e^{\text{closing}} \mid \theta_1 < |x_1| \leq \theta_2\}, & \text{—închisII} \\ \Omega_4 := \{x \in \mathbb{R}^5 \mid x_3 > \omega_e^{\text{closing}} \mid \theta_2 < |x_1|\}, & \text{—închisIII} \end{array} \right. \quad (\text{I.41})$$

unde regiunea Ω_1 corespunde modului deschis, în timp ce regiunile $\Omega_2, \Omega_3, \Omega_4$ corespund celor trei faze ale modului închis. $\omega_e^{\text{closing}}$ este viteza de prag de închidere a motorului, iar θ_1, θ_2 sunt valori de prag ale unghiului de torsiune dintre motor și transmisie, folosite pentru comutarea între modurile de lucru ale ambreiajului. Logica de comutare este ilustrată în Fig. I.14 și se poate observa că ambreiajul rămâne în modul deschis atât timp cât viteza motorului nu atinge viteza de închidere. Atunci când această viteză este atinsă, comutarea între cele trei faze ale modului închis se face în funcție de valoarea primei variabile de stare, unghiul de torsiune dintre motor și transmisie. Când acest unghi este mai mic decât valoarea de prag θ_1 , sistemul se află în prima fază a modului închis. Atunci când unghiul

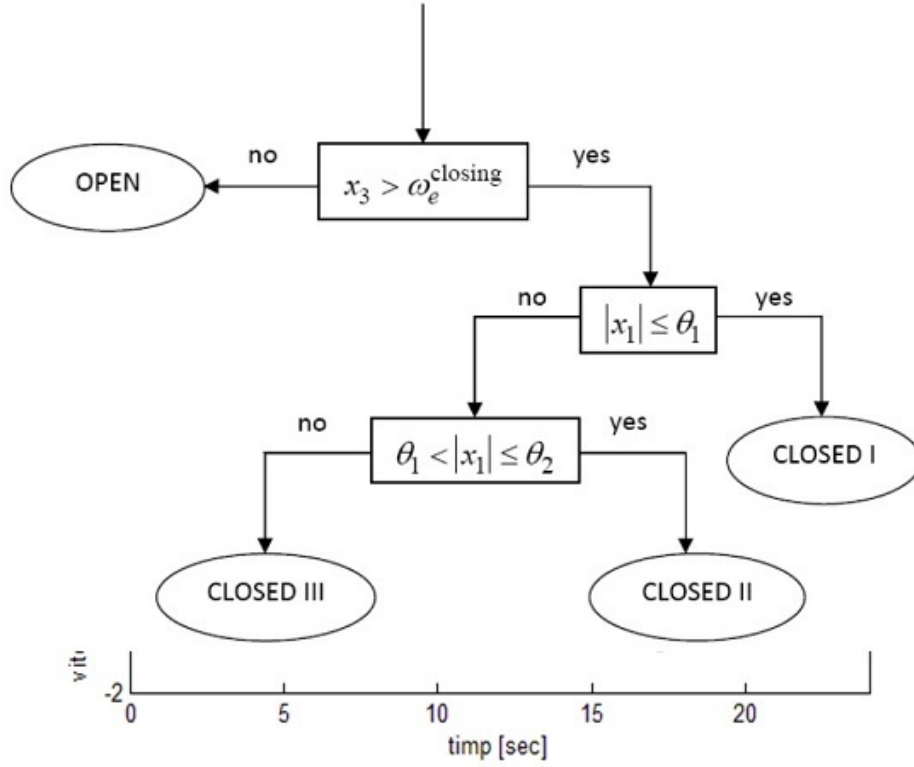


Figura I.14: Logica de comutare a ambreiajului.

trece de valoarea de prag θ_1 dar este mai mic decât valoarea de prag θ_2 , sistemul se află în cea de-a doua fază a modului închis. Atunci când unghiul trece de valoarea de prag θ_2 , sistemul se află în cea de-a treia fază a modului închis.

Atunci când are loc o tranziție din modul deschis în modul închis, trebuie impusă următoarea condiție de resetare:

$$\forall t_1 \in \mathbb{R}_+, \forall t_2 \in \mathbb{R}_{>t_1}, \text{if } x(\tau) \in \Omega_1, \forall \tau \in \mathbb{R}_{[t_1, t_2)} \text{ and } x(t_2) \in \Omega_2, \text{ set } x_1(t_2) := 0. \quad (\text{I.42})$$

Noul model are matricile sistemului $A_{c1}, A_{c2}, A_{c3}, A_{c4}$, corespunzătoare modurilor de lucru ale ambreiajului:

$$A_{ci} = \begin{pmatrix} 0 & 0 & 1 & -i_t & 0 \\ 0 & 0 & 0 & \frac{1}{i_f} & -1 \\ -\frac{k_{ci}}{J_1} & 0 & \frac{-D_{sum1}}{J_1} & \frac{d_{ci}i_t}{J_1} & 0 \\ \frac{k_{ci}i_t}{J_2} & -\frac{k_d}{i_f J_2} & \frac{d_{ci}i_t}{J_2} & \frac{-D_{sum2}}{J_2} & \frac{d_d}{i_f J_2} \\ 0 & \frac{k_d}{J_3} & 0 & \frac{d_d}{i_f J_3} & \frac{-d_{wheel}}{J_3} \end{pmatrix}, \quad (\text{I.43})$$

cu $D_{sum1} = d_{ci} + d_e$, $D_{sum2} = d_{ci}i_t^2 + d_2 + \frac{d_d}{i_f^2}$, $d_{wheel} = d_w + d_d + c_{r2}$ și coeficientul de elasticitate k_{ci} și coeficientul de amortizare al ambreiajului d_{ci} .

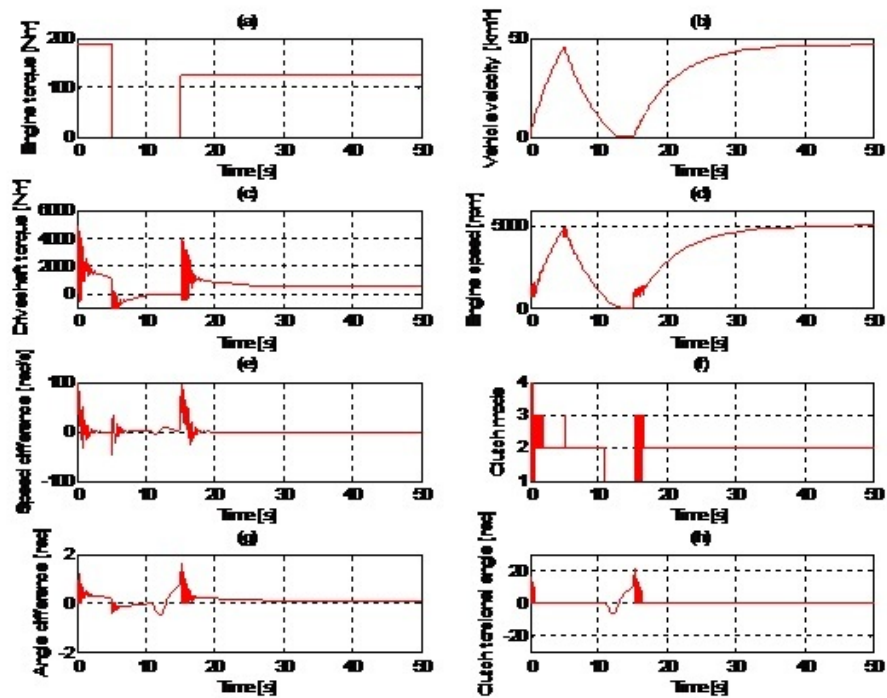


Figura I.15: Rezultate de simulare.

I.2.2.3 Implementarea modelului în Matlab/Simulink și rezultate de simulare

În Fig. I.15 pot fi observate rezultatele obținute în urma simulării modelului prezentat în subcapitolul anterior: a) cuplul motor (mărimea de intrare); b) viteza de deplasare a autovehiculului (mărimea de ieșire); c) cuplul de torsiune al osiei; d) viteza motorului; e) diferența dintre viteza motorului și viteza roților; f) modul de lucru al ambreiajului; g) diferența dintre unghiul motorului și unghiul cutiei de viteze; h) unghiul de torsiune al ambreiajului. În subfigurile d), f) și h) sunt puse în evidență mărimile din model influențate ce acționarea ambreiajului. Parametrii utilizați în simulare pot fi regăsiți în lucrarea [57].

I.2.3 Modele roboți mobili

Roboții mobili pot fi caracterizați de modele foarte simple, de tipul elementelor de ordin I cu întârziere. Astfel, în [13, 58, 59] au fost dezvoltate modele pentru roboții mobili de tip mOway și SRV.

I.2.3.1 Modelarea roboților mobili mOway

Roboții mobili de tip mOway (Fig. I.16) sunt foarte utili în demonstrarea teoriilor referitoare la plutoanele de autovehicule datorită costurilor reduse de achiziție și a ușurinței de programare și interfațare.



Figura I.16: Roboții mobili mOway.

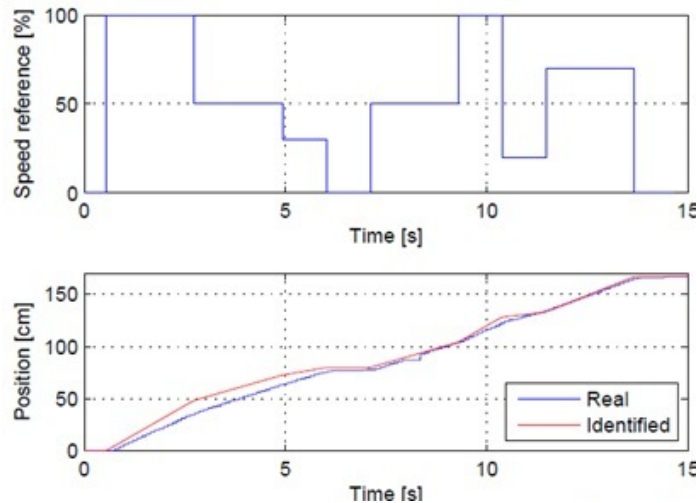


Figura I.17: Experiment identificare.

Modelul robotului a fost determinat utilizând răspunsul obținut în urma aplicării la intrarea acestuia a unei secvențe de identificare de tip treaptă multiplă, experiment ilustrat în Fig. I.17. Funcția de transfer rezultată între viteza de referință și poziția robotului este dată de [13,58]:

$$G(s) = \frac{0.22}{s(0.04s + 1)}. \quad (\text{I.44})$$

Comparația între ieșirea reală a robotului (poziția) și rezultatul modelului identificat este ilustrat în Fig. I.17, corelarea dintre cele două semnale fiind de 94%. A se reține că de intrarea procesului nu este o forță, ci o referință pentru viteza robotului dată ca procent. Totodată, viteza este măsurată în cm/s și are un maxim de 22 cm/s rezultând un factor de amplificare pentru funcția de transfer egal cu 0.22.

I.2.3.2 Modelarea roboților mobili SRV

Roboții, bazați pe platforma Surveyor SRV-1, sunt de dimensiunea unei palme. Roboții sunt foarte simpli ca structură, fiecare dintre ei fiind în măsură să urmeze un alt robot utilizând localizarea vizuale și controlul vitezei.

Un experiment similar celui utilizat pentru determinarea modelului roboților mOway a fost realizat și pentru acest tip de roboți, rezultând un model de tip funcție de transfer

rezultată între viteza de referință și poziția robotului este dată de [59]

$$G(s) = \frac{1.24}{s(0.1s + 1)}. \quad (\text{I.45})$$

I.2.4 Modele ale plutoanelor de autovehicule

În literatura de specialitate există trei tipuri de modele considerate pentru plutoanele de vehicule: modele centralizate foarte complexe pentru întregul pluton de vehicule [23], modele descentralizate în care dinamicile vehiculelor sunt decuplate, dar plutonul are cuplaje între vehicule [5, 25–29, 60, 61], și modele distribuite în care vehiculele din pluton sunt modelate considerând cuplaje-le dintre ele [7, 8], dar niciunul dintre aceste modele nu consideră restricțiile de comunicație. Ar fi util un model distribuit mai realist ce consideră cuplajele dintre vehicule și care include restricțiile de comunicație pentru îmbunătățirea performanțelor. În ceea ce privește regulatoarele existente, în literatura de specialitate sunt propuse mai multe tipuri: regulator bazat pe mașină de stări [24], regulator ajustabil [27], regulator liniar pătratic [25], regulator robust cu parametri liniar variabili [26] și regulatoare predictive [23, 60, 61], dar niciunul dintre acestea nu consideră restricțiile hardware/computaționale pentru implementarea lor în sisteme încorporate. Dintr-un alt punct de vedere regulatoarele de mai sus pot fi clasificate în două tipuri: regulatoare centralizate [23] și regulatoare descentralizate [24–29, 60, 61]; *limitările abordărilor curente sunt evident legate de neluarea în considerare a cuplajelor dintre vehiculele din pluton.*

I.2.4.1 Configurația unui pluton și modelarea dinamicii vehiculului

Un pluton de vehicule echipate cu sisteme CACC este descris în Fig. I.18. Stările de deplasare a al i -lea vehicul includ viteza și accelerația acestuia. Plutonul considerat este unul eterogen deoarece este alcătuit din autovehicule cu dimensiuni și caracteristici diferite de la mașini mici de pasageri până la camioane. Poziția barei din spate a vehiculului lider este notată cu s_i . Distanța de siguranță a al i -lea vehicul din pluton este notată cu d_i . Din configurația plutonului se poate determina eroarea de distanță i cu expresia [25]:

$$\delta_i(k) = s_{i-1}(k) - s_i(k) - d_i - l_i. \quad (\text{I.46})$$

Tehnologia cheie a plutonului cu sisteme CACC este reprezentat de comunicațiile wireless folosite pentru transportul informațiilor de-a lungul lanțului de vehicule astfel încât fiecare vehicul să primească date de la un anumit număr de vehicule care circulă înaintea lui. În acest caz există trei tipuri de comunicații:

- strategia lider-predecesor-următor;
- strategia de tipul predecesor-următor;
- omunicațiile în întregul pluton.

Strategia considerată în lucrarea [25] este aceea în care vehiculul gazdă poate primi informații despre vehiculul cel mai apropiat și despre lider. Cunoscând aceste informații, regulatorul va avea caracteristici predictive astfel comportamentul vehiculelor poate deveni

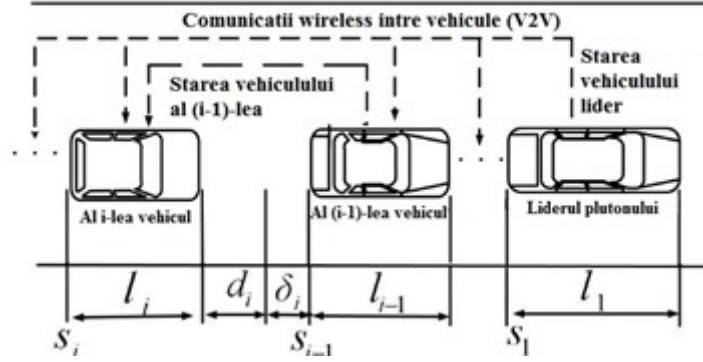


Figura I.18: Structura unui pluton.

și mai inteligent. Diverse modele pentru dinamica vehiculului au fost folosite pentru studiul controlului longitudinal al plutoanelor. Pentru un vehicul care se deplasează cu o direcție și o viteză constantă, se presupune că al i -lea vehicul dintr-o formație închisă ce este alcătuită din n vehicule poate fi reprezentat de următorul model liniar spațial cu trei stări:

$$\begin{aligned} \dot{x}_i(t) &= \Phi_i x_i(t) + \Pi_i u_i(t), \\ \Phi_i &= \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -1/\tau_i \end{bmatrix}, \quad \Pi_i = \begin{bmatrix} 0 \\ 0 \\ 1/\tau_i \end{bmatrix}, \\ x_i(t) &= [s_i(t) \quad v_i(t) \quad a_i(t)]^T, \end{aligned} \quad (\text{I.47})$$

unde v_i și a_i sunt viteza și accelerația celui de-al i -lea vehicul; $u_i \in \mathbb{R}^1$ reprezintă intrarea de control; $x_i \in \mathbb{R}^3$ este starea sistemului pentru al i -lea vehicul, τ_i este timpul constant cauzat de sistemul de propulsie al vehiculului care reprezintă caracteristicile defertelor vehicule. Considerând că de obicei vehiculul este proiectat și implementat în domeniul discret de timp, relația continuă în timp (I.47) este convertită într-un model discret prin modul de discretizare ZOH (zero-order hold), obținându-se:

$$x_i(k+1) = A_i x_i(k) + B_i u_i(k), \quad (\text{I.48})$$

unde k reprezintă al k -lea punct de eșantionare, A_i și B_i sunt matrici sistem, exprimate astfel:

$$A_i = \sum_{k=0}^{\infty} \frac{\Phi^k T_s^k}{k!}, \quad B_i = \sum_{k=0}^{\infty} \frac{\Phi^{k-1} T_s^k}{k!} \Pi, \quad (\text{I.49})$$

unde T_s este perioada de eșantionare.

După definirea modelului și configurației plutonului de autovehicule se pot stabili obiectivele necesare proiectării sistemelor de control pe baza unor cerințe referitoare la deplasarea în pluton [25]:

1. Pe baza unor trăsături a unui vehicul gazdă, condiția suprafeței drumului, și situațiile din trafic reale, strategia potrivită între vehicule ar trebui selectată de către șofer.

2. Starea de echilibru a erorii de distanță ar trebui să fie egală cu zero pentru toate vehiculele din pluton astfel încât să fie asigurată capacitatea de urmărire între vehicule.
3. Când vehiculul lider accelerează, vehiculul gazdă ar trebui să accelereze înainte să detecteze că cel mai apropiat vehicul accelerează și el, pentru a evita creșterea distanței între vehicule.
4. În momentul în care liderul frânează, vehiculul vizat va trebui să decelereze înainte de detectarea faptului că vehiculul precedent decelerează pentru a evita coliziunea între acestea.

Obiectivele 2, 3 și 4 pot fi văzute ca și capacitatea de urmărire a vehiculului gazdă. Pentru a satisface obiectivele 3 și 4, viteza și accelerația dorite ale vehiculului vizat vor trebui proiectate pe baza stărilor de deplasare care fac referire la liderul plutonului și la cel mai apropiat vehicul din față.

I.2.4.2 Modelarea unui pluton de vehicule cu ACC centralizat

În [62] este propusă o metodă de dezvoltare a unui model liniar pentru plutoane alcătuite din n vehicule urmăritoare și un lider. Acest model ar trebui să conțină comportamentul longitudinal al vehiculelor și ar trebui să furnizeze integral și comportamentul stabilității de bază, care poate fi atins, dacă este permisă o controlabilitate totală a sistemului. În acest caz vehiculul care conduce plutonul este controlat manual sau prin alte sisteme de asistență a șoferului, dar nu prin sisteme ACC. Se contrazice astfel controlabilitatea totală a sistemului. O soluție pentru a satisface cele două presupuneri este considerarea doar a valorilor relative între două vehicule consecutive. Atât timp cât valorile relative ale vehiculului lider condus manual rămân controlabile acesta poate face parte din sistem.

Se consideră o reducere a modelelor vehiculelor la parametrii ce definesc poziția, viteza, accelerația și gradul de înregistrare a sacadării. În cazul prezentat un pluton este alcătuit din n vehicule V_1, \dots, V_n și un vehicul lider V_0 . Pentru al i -lea vehicul din pluton V_i , se cunoaște următoarea ecuație de mișcare:

$$d_i(t) = x_i^{(r)} + v_i^{(r)}t + \frac{1}{2}a_i^{(r)}t^2 + \frac{1}{6}(j_{i-1} - j_i)t^3, \quad (\text{I.50})$$

unde t este timpul, $d_i(t)$ este distanța care depinde de momentul de timp t , iar la $t = 0$ se cunosc distanța $x_i^{(r)} = x_{i-1} - x_i$, viteza relativă $v_i^{(r)} = v_{i-1} - v_i$ și accelerația relativă $a_i^{(r)} = a_{i-1} - a_i$ între V_i și V_{i-1} cu notația (r) care definește o valoare relativă. Măsura mișcărilor sacadate (jerk) ale vehiculului V_i este notată cu j_i . Se ia în considerare că $j_0 \equiv 0$ cât timp vehiculul lider V_0 nu este controlabil printr-un sistem ACC. Se va obține modelarea plutonului ca un lanț în care fiecare membru este conectat cu predecesorul său, dacă vor fi luate în considerare valorile relative ale mărimilor amintite anterior pentru reprezentarea în spațiul stărilor. În cazul în care unul dintre vehicule a avut vreo eroare de comunicație, acest lanț se va rupe. Pentru a evita dezavantajele generate de acest neajuns, s-au creat conexiuni suplimentare prin definirea următoarelor valori de stare:

$$d_i^w(t) := \sum_{l=1}^i w_l d_{i-l+1}(t). \quad (\text{I.51})$$

Au fost introduse astfel ponderile pozitive w_1, \dots, w_n , care pot fi alese în mod arbitrar. În cazul prezentat s-au ales ponderi monotonic descrescătoare. În concordanță cu acestea s-au ales vitezele relative $d_i^{w'}(t)$ și accelerațiile relative $d_i^{w''}(t)$ pentru $i = 1, \dots, n$.

Se obține reprezentarea discretă a sistemului prin înlocuirea timpului t cu kT_s , unde $k > 0$ și T_s este perioada de eșantionare. Reprezentarea discretă a plutonului în spațiul stărilor este descrisă de:

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k), \\ y(k) &= x(k), \end{aligned} \quad (\text{I.52})$$

cu vectorul de stări:

$$x(k) := \begin{pmatrix} d_n^{w'}(k) \\ d_n^{w''}(k) \\ j_n(k) \\ \vdots \\ d_1^{w'}(k) \\ d_1^{w''}(k) \\ j_1(k) \end{pmatrix} \in^{4n}. \quad (\text{I.53})$$

Pentru a se obține comportamentul integral al regulatorului, gradul de înregistrare a mișcării sacadate nu este considerat ca o valoare de intrare, dar în schimb se definește $u_i(k) := j_i(k+1) - j_i(k)$ pentru $i = 1, \dots, n$.

Pe lângă cele definite anterior în ecuațiile (I.53) mai există vectorul ieșirilor $y(k)$ și matricile A și B cu expresiile:

$$A := \begin{pmatrix} A_n & A_{n-1} - A_n & \dots & A_1 - A_2 \\ & \ddots & \ddots & \vdots \\ & & A_n & A_{n-1} - A_n \\ & & & A_n \end{pmatrix} \in^{4n \times 4n}, \quad (\text{I.54})$$

$$B := \begin{pmatrix} B_1 & & \\ & \ddots & \\ & & B_1 \end{pmatrix} \in^{4n \times n}, \quad (\text{I.55})$$

în care A_i și B_1 au următoarele expresii:

$$A_i := \begin{pmatrix} 1 & T_s & \frac{1}{2}T_s^2 & -\frac{1}{6}w_iT_s^3 \\ 0 & 1 & T_s & -\frac{1}{2}w_iT_s^2 \\ 0 & 0 & 1 & -w_iT_s \\ 0 & 0 & 0 & 1 \end{pmatrix}, \quad (\text{I.56})$$

$$B_1 := \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix}. \quad (\text{I.57})$$

Pentru a garanta siguranța și confortul acestui sistem se introduc constrângeri pentru decalajul dintre două vehicule consecutive și pentru viteza, accelerația și mișcarea sacadată ale tuturor vehiculelor din pluton. Constrângerile echivalente pentru $d_i^w(t)$, $d_i^{w'}(t)$ și $d_i^{w''}(t)$ pot fi atinse iterativ, pornind de la $i \equiv 1$. Toate constrângerile amintite pot fi rezumate ca $x_{\min} \leq x \leq x_{\max}$ sau $x := \{x | x_{\min} \leq x \leq x_{\max}\}$.

Trebuie să se ia în considerare că:

- constrângerile sunt independente de viteza v_0 și accelerația a_0 ale vehiculului lider;
- se poate obține foarte ușor mărghinirea prin introducerea unei limite superioare mari pentru $d_i(k)$;
- prin transformarea adecvată a vectorului de stări x , se poate obține $0 \in \text{int}\mathbb{X}$.

I.3 (A1.2) Modelarea întârzierilor variabile în timp din rețea folosind aproximări politopice sau considerându-le ca perturbații pentru a le exprima ca partiții poliedrale ale spațiului stărilor

I.3.1 Modelarea întârzierilor folosind aproximări politopice

Cunoscând limita superioară determinată în prealabil a întârzierilor variabile în timp introduse de rețeaua CAN, sarcina este de a se obține un model discret al buclei în circuit închis care ține cont de întârzierile care apar în rețea în limitele determinate. Astfel, se consideră sistemul continuu, cu întârziere pe calea directă introdusă de rețeaua de comunicații [34]

$$\begin{aligned} \dot{x}(t) &= A_c x(t) + B_c u(t - \tau), \\ u(t) &= u_k, \forall t \in [t_k + \tau_k, t_{k+1} + \tau_{k+1}), \end{aligned} \quad (\text{I.58})$$

unde $x(t)$ este starea sistemului, $u(t)$ este intrarea sistemului, A_c și B_c sunt matricile sistemului, $\tau = \tau^{sc} + \tau^{ca}$ reprezintă întârzierea introdusă de rețea, $t_k = kT_s$, $k \in \mathbb{Z}_+$, $T_s \in \mathbb{R}_+$ este perioada de eșantionare și se presupune că $u(t) = u_0$ pentru toți $t \in [0, \tau_0]$ cu $u_0 \in \mathbb{R}^m$ un vector constant determinat a priori. $u_k \in \mathbb{R}$ este comanda generată la momentul de timp $t = t_k$. $\tau_k \in \mathbb{R}_{[0, \tau^{max}]}$ reprezintă întârzierea introdusă de rețea la momentul de timp $k \in \mathbb{Z}_+$ și $\tau^{max} \in \mathbb{R}_+$ este întârzierea maximă care poate fi introdusă de rețeaua de comunicații.

Se consideră întârzierea maximă de forma $\tau^{max} = (\Upsilon + v)T_s$, unde $\Upsilon \in \mathbb{Z}_{\geq 1}$ și $v \in \mathbb{R}_{[0, 1]}$. Utilizând aproximarea politopică descrisă în [63], modelul discret este inclus în

$$x_{k+1} \in \phi(x_k, \mathbf{u}_{[k-\Upsilon-1, k]}), \quad k \in \mathbb{Z}_+, \quad (\text{I.59})$$

unde intrările $\mathbf{u}_{[-\Upsilon-1, 0]}$ au valori predeterminate și

$$\begin{aligned} \phi(x_k, \mathbf{u}_{[k-\Upsilon-1, k]}) &:= \{A_d x_k + B_d u_k + \sum_{i=0}^{\Upsilon} \Delta_i (u_{k-i-1} - u_{k-i}) \mid \\ &| \Delta_i \in \Delta^{T_s}, i \in \mathbb{Z}_{[0, \Upsilon-1]}, \Delta_{\Upsilon} \in \Delta^{vT_s}\}. \end{aligned} \quad (\text{I.60})$$

I.3.2 Modelarea întârzierilor de intrare și de ieșire ca perturbații

Se consideră un model discretizat al unui proces fizic de forma

$$x_{k+1} = A_d x_k + B_d u_k, \quad (\text{I.61})$$

unde x_k este starea sistemului, u_k este mărimea de comandă, iar A_d și B_d sunt matricile sistemului.

I.3.2.1 Întârzieri pe calea directă

Scopul este de a găsi un set mărginit \mathbb{W}_u , în care să fie incluse toate perturbațiile posibile care pot să apară din cauza întârzierilor variabile în timp, introduse de rețeaua de comunicații pe calea directă, știind că mărimea de intrare a procesului fizic este mărginită. Metoda propusă este diferită de cea prezentată în [64,65], în care, un observator este utilizat pentru a determina diferența dintre semnalul de comandă calculat și semnalul de comandă care a produs ieșirea măsurată.

Se consideră că mărimea de comandă este mărginită astfel

$$u^{\min} \leq u_k \leq u^{\max}, \quad (\text{I.62})$$

unde u^{\min} și u^{\max} sunt valorile minime, respectiv maxime, ale mărimii de comandă care este furnizată ca semnal de intrare procesului fizic. Apoi, perturbația introdusă de rețeaua de comunicații poate fi mărginită astfel [66]

$$u^{\min} - u^{\max} \leq u_k^d \leq u^{\max} - u^{\min}. \quad (\text{I.63})$$

Mai mult, dacă incrementul mărimii de control discrete este mărginit

$$-u^\Delta \leq \Delta u_k \leq u^\Delta, \quad (\text{I.64})$$

unde $\Delta u_k := u_k - u_{k-1}$, pentru toți $k \in \mathbb{Z}_{\geq 1}$, cu u_0 o valoare predeterminată și u^Δ este valoarea maximă a incrementului la fiecare perioadă de eșantionare $k \in \mathbb{Z}_{\geq 1}$, perturbația poate fi remărginită astfel [66,67]

$$-\bar{a}u^\Delta \leq u_k^d \leq \bar{a}u^\Delta. \quad (\text{I.65})$$

I.3.2.2 Întârzieri pe calea de reacție

Scopul este de a găsi un set mărginit \mathbb{W}_x , în care să fie incluse toate perturbațiile posibile care pot să apară din cauza întârzierilor variabile în timp, introduse de rețeaua de comunicații pe calea de reacție, știind că mărimea de intrare a procesului fizic este mărginită.

Fie modelul discret al unui proces fizic de forma

$$x_{k+1} = A_d x_k + B_d (u_k + u_k^d), \quad (\text{I.66})$$

unde A_d și B_d reprezintă matricile discretizate ale sistemului. Atunci, perturbația pe calea de reacție poate fi modelată astfel [35]:

$$\begin{aligned} x_k^d &= x_{k-a_k-b_k} - A_d^{b_k} x_{k-a_k-b_k} - \sum_{i=0}^{b_k-1} A_d^i B_d (u_{k-a_k-i-1} + u_{k-a_k-i-1}^d) = \\ &= (I_n - A_d^{b_k}) x_{k-a_k-b_k} - \sum_{i=0}^{b_k-1} A_d^i B_d (u_{k-a_k-i-1} + u_{k-a_k-i-1}^d), \end{aligned} \quad (\text{I.67})$$

unde a_k și b_k reprezintă valoarea întârzierilor de pe calea directă și de pe calea de reacție de la momentul de timp $k \in \mathbb{Z}_+$ exprimate ca un număr de perioade de eșantionare.

$x_{k-a_k-b_k}$ este cunoscut la momentul de timp k , toate u_{k-i} , $i \in \mathbb{Z}_{\geq 1}$ sunt cunoscute, toate u_{k-i}^d , $i \in \mathbb{Z}_{\geq 1}$ sunt mărginite, a_k și b_k sunt mărginite de \bar{a} și \bar{b} , respectiv, deci x_k^d poate fi mărginit în mod dinamic la fiecare perioadă de eșantionare $k \in \mathbb{Z}_+$.

Astfel, (I.67) devine

$$x_{k+1} = A_d x_k + B_d u_k + B_d u_k^d + x_k^d. \quad (\text{I.68})$$

Deși întârzierile variabile în timp cauzează o perturbație variabilă în timp, seturile \mathbb{W}_u , care este definit de $B_d u_k^d$, cu u_k^d din (I.63) sau din (I.65), și \mathbb{W}_x , care este definit de x_k^d din (I.67) rămân fixate. Astfel, această metodă de modelare este potrivită pentru utilizarea rezultatelor prezentate în [68], în care perturbațiile sunt luate în considerare în mod explicit în faza de proiectare a regulatorului predictiv.

I.4 (A1.3) Modelarea restricțiilor hardware ca partiții poliedrale ale spațiului stărilor

I.4.1 Implementarea reguletoarelor predictive în timp real

O etapă importantă a proiectului constă în dezvoltarea de programe de timp real pentru implementarea algoritmilor de control predictivi distribuți multiagent pe sisteme încorporate. Acești algoritmi vor fi testați experimental în laborator folosind un pluton de vehicule (pluton de roboți mobili). Deși controlul predictiv bazat pe model reprezintă una dintre metodele de control avansate folosite în industrie, el nu este totuși folosit în sistemele încorporate ce sunt critice din punctul de vedere al siguranței, cum ar fi cele din domeniul medical, spațial, al roboticii sau al industriei autovehiculelor. Resursele limitate ale sistemelor încorporate împreună cu necesitatea unor performanțe predictibile limitează înlocuirea algoritmilor de control convenționali folosiți în industrie cu reguletoare predictive implementate cu sisteme încorporate. Dificultatea acestei etape a proiectului este dată de fapt de dificultatea implementării unor reguletoare predictive care să respecte cerințele de timp impuse de către aplicația propriu-zisă. Dat fiind că algoritmii reguletoarelor predictive impun optimizări numerice ce necesită calcule intensive, ce se bazează pe acuratețea aritmeticii în virgule mobile, performanța acestora va fi influențată în mod negativ de puterea limitată de calcul și de capacitatea limitată de memorie a anumitor sisteme încorporate.

I.4.1.1 Arhitectura hardware folosită

Un pas important este reprezentat de identificarea unor arhitecturi hardware care să permită sistemelor încorporate să implementeze cu succes regulatoare predictive. Regulatele predictive încorporate se regăsesc în forme multiple: regulatoare industriale, regulatoare logice programabile (Programmable Logic Controller - PLC) și unități electronice de control (Electronic Control Unit - ECU). Regulatele industriale sunt unități de procesare cu coprocesoare în virgulă mobilă ce dispun de cantități mari de memorie de date și de program. Sunt regulatoare standardizate proiectate să fie folosite în sisteme distribuite de control [69]. Regulatele logice programabile sunt asemănătoare cu regulatele industriale, dar sunt proiectate să fie mai robuste decât acestea. Ele prezintă procesoare cu capacitate de procesare și memorare limitată și doar modelele avansate pot prezenta coprocesoare în virgulă mobilă. În [70] s-a arătat că implementarea regulatelor predictive cu ajutorul regulatele logice programabile este uneori dificilă deoarece în majoritatea cazurilor acestea din urmă nu permit rularea de cod C/C++ realizat și testat pe PC-uri. Limbajele de programare sunt diferite, implementarea unui regulator predictiv pe un PLC presupunând traducerea programului realizat pe PC în limbajul suportat de PLC.

Foarte atractive pentru realizarea regulatelor predictive încorporate sunt unitățile electronice de control ce sunt proiectate special pentru aplicații și conțin microcontroller-e sau circuite integrate digitale configurabile (Field Programmable Gate Array - FPGA).

Implementarea regulatelor predictive cu FPGA

Implementarea regulatelor predictive cu FPGA a fost studiată încă din 2006, când în [71] s-a studiat modul în care regulatele predictive pot fi realizate cu ajutorul unui procesor și a unui coprocesor, ambele dispozitive fiind sintetizate și încărcate în FPGA. Procesorul luat în calcul este microcontroller-ul pe 32 de biți MPC555 de la Motorola ce conține o unitate de virgulă mobilă pe 64 de biți. Marea majoritate a operațiilor aritmetice sunt realizate de către coprocesor, acest lucru permițând procesorului să acționeze ca un dispozitiv de tip master și să execute și alte task-uri. Implementarea regulatelor predictive cu ajutorul FPGA este studiată și în [72], arătându-se că dispozitivele de tip FPGA permit creșteri de viteză semnificative față de implementările uzuale cu microcontroller-e sau cu procesoare numerice de semnal (DSP), acest lucru datorându-se paralelismului ce poate fi implementat cu ajutorul FPGA-urilor. Regulate predictive liniare ce au rate de eșantionare de ordinul milisecundelor și se bazează pe paralelismul oferit de dispozitivele FPGA au fost realizate de [73] și [74]. Un regulator bazat pe programare pătratică ce profită de eficiența și flexibilitatea FPGA-urilor este prezentat în [75], iar în [76] este trecut în revistă un regulator predictiv ce prezintă frecvențe de eșantionare de 1MHz.

Implementări ale regulatelor predictive cu număr mare de intrări și cu aplicații în industria aerospațială sunt prezentate în [77–79]. Cercetările au continuat și în direcția obținerii de regulate predictive de viteză ridicată, rezultatele fiind prezentate în [80]. De asemenea, un regulator ce rezolvă problema programării pătratice prin metoda QPKWIK este prezentat în [81].

Implementarea regulatelor predictive cu microcontrolere

Dispozitivele FPGA permit implementarea de reglatoare predictive cu frecvențe mari de eșantionare datorită frecvențelor lucru de până la 300MHz. Cu toate acestea, printre dezavantajele reglatoarelor implementate cu FPGA se numără și consumul ridicat. Cum una dintre prioritățile sistemelor încorporate este aceea de a putea rula cu un consum cât mai redus de energie, un foarte mare interes a fost acordat studiului reglatoarelor predictive bazate pe microcontroller-e. În [82, 83] sunt prezentate soluții dezvoltate cu microcontroller-ul pe 32 de biți STM32F103xB ce încorporează nucleul RISC ARM Cortex-M3 de la STMicroelectronics ce lucrează la frecvențe de maxim 72MHz.

Eforturile facute în direcția implementării unui regulator predictiv pentru un sistem pneumatic sunt prezentate în [84]. Microcontroller-ele testate au fost variantele pe 32 de biți de la Atmel AT91SAM7256 cu nucleu ARM7 și AT91SAMG20 cu nucleu ARM9, pentru ultimul model obținându-se frecvențe maxime de eșantionare de 800Hz. În [85] este prezentată o metodă de sintetizare a regulatorului predictiv cu ajutorul mediului Matlab pentru a obține reglatoare rapide, cu frecvențe de eșantionare de domeniul kHz-ilor. Tot cu microcontroller cu nucleu ARM7 (Atmel AT91SAM7S256) este proiectat și regulatorul prezentat în [38]. Implementări ale reglatoarelor predictive bazate pe microcontroller-e mai puțin puternice (8 și 16 biți) sunt prezentate în [86] și respectiv [87], microcontroller-ele folosite fiind ATmega2560 de la Atmel și respectiv MSP430 de la Texas Instruments. În [86] autorul specifică faptul că frecvența regulatorului este de doar 50Hz și că utilizarea unui microcontroller mai puternic (pe cel puțin 16 biți) este recomandabilă.

În ultimii ani subiectul aeronavelor fără pilot (engl. Unmanned Aerial Vehicle - UAV) a devenit actual și a captat foarte mult interes. În comunitatea academică au fost depuse tot mai multe eforturi în direcția identificării metodelor de control ce ar putea fi aplicate pentru acest tip de vehicule. Pe lângă reglatoarele clasice, reglatoarele predictive au fost implementate și testate pe o gamă largă de microcontrollere [88–90].

Bouffard a realizat controlul unui quadrotor cu ajutorul unui regulator PID și a unui regulator predictiv. El a ales să folosească regulatorul PID pentru controlul altitudinii și regulatorul predictiv pentru controlul deplasării pe laterală [88]. Regulatorul predictiv a fost proiectat cu ajutorul instrumentului Multi-Parametric Toolbox din Matlab și a fost rulat pe un procesor ATOM. Bangura și Mahony au inclus și ei un regulator predictiv în controlul unui quadrotor. Abordarea lor combină paradigma standard de control ierarhic în care controlul este separat în controlul de bază al motoarelor, controlul intermediar al comportării dinamice și controlul de nivel înalt al traiectoriei, cu o strategie de control predictiv [89]. Structura propusă astfel a condus la algoritmul de control predictiv pentru un model de complexitate redusă, algoritm ce este implementat în timp real pe sistemul încorporat. Frecvența de lucru obținută pentru regulatorul predictiv este de 16Hz, ea fiind limitată de standardul de comunicație radio folosit.

Baca și-a concentrat cercetările asupra reglatoarelor predictive utilizate pentru controlul unui tricopter. Tot el dă un alt argument în favoarea utilizării controlului predictiv pe microcontrollere, el evidențiind faptul că aceasta, odată realizată, va permite crearea de micro-vehicule aeriene (engl. Micro Aerial Vehicle - MAV) precum și crearea de roiuri de MAV ce ar putea rezolva sarcini de detectare și localizare de la distanță a obiectelor periculoase precum și recunoaștere a unui mediu necunoscut. În [90] este prezentată o variantă particularizată a unui regulator predictiv cu programare pătratică ce permite controlul

tricopter-ului cu un orizont de predicție de 2,2 secunde. Baca a distribuit sarcinile de procesare pe două microcontroller-e. Primul dintre ele, un microcontroller pe 8 biți de la Atmel (ATxMega128a3u), ce lucrează la frecvența de 32MHz și are o memorie SRAM de 8kB, este însărcinat cu comunicațiile și cu eventuale task-uri minore. El comunică prin interfața serial UART cu 3 componente din sistemul încorporat. Cel de-al doilea microcontroller, STM32F415RGT6, lucrează pe 32 de biți și este produs de către STMicroelectronics. Este construit pe un nucleu ARM Cortex M4 cu unitate de procesare în virgulă mobilă, lucrează la frecvența de 168MHz și oferă 192kB de RAM. El implementează regulatorul predictiv și un filtru Kalman, fără a fi conectate la el alte dispozitive periferice.

Regulatele predictive bazate pe logică fuzzy reprezintă de asemenea un punct de interes. Microcontroller-ul utilizat de [91] este LM4F120H5QR, un microcontroller pe 32 de biți bazat pe un nucleu ARM Cortex M4F de la Texas Instruments, ce rulează la frecvența de 80MHz și dispune de 32kB de memorie SRAM, 256kB memorie program și 2kB memorie EEPROM. Dată fiind dimensiunea relativ mică a memoriei SRAM, parametrii modelului au fost calculați offline.

Marea majoritate a cercetărilor sunt orientate în direcția implementării regulatele predictive pe platforme hardware ce conțin unități de procesare în virgulă mobilă. O abordare diferită o au [92], ce demonstrează că, în anumite condiții, trecerea de la calcule în virgulă mobilă la cele în virgulă fixă se poate face fără o pierdere de calitate a algoritmului de control. Autorii subliniază că, deși abordarea nu este simplă și nici directă, ea asigură o reducere a cerințelor de resurse și a întârzierilor cauzate de calculele aritmetice cu cel puțin un ordin de mărime, putându-se obține, pentru particularizări ale aplicației, reduceri de până la două ordine de mărime.

După cum se poate observa în [90], sistemele incorporate ce implementează și regulatele predictive sunt sisteme complexe, iar utilizarea sistemelor de operare de timp real (SOTR) permit dezvoltarea mult mai simplă a aplicațiilor de timp real critice, complexe. Există o gamă diversă de SOTR ce pun la dispoziție obiecte și servicii API pentru gestionarea execuției multitasking, sincronizarea și comunicarea dintre procese, partajarea resurselor pasive. Dezavantajele principale legate de folosirea SOTR constă în creșterea consumului de memorie RAM/ROM și în utilizarea a aproximativ 2-4% din capacitatea de procesare a microcontroller-ului pentru execuții multitasking, în mod preventiv. De aceea, în funcție de necesitățile sistemului încorporat dezvoltat, SOTR ar trebui ales ținând cont și de amprenta de memorie caracteristică, cerințele hardware minimale impuse și tipul aplicațiilor de timp real acceptate (e.g. critice sau necritice, număr maxim de task-uri permise, etc.).

După cum am prezentat anterior, Baca folosește două microcontroller-e în cadrul sistemului încorporat folosit pentru controlul tricopter-ului. Pe ambele microcontroller-e rulează aplicații ce au la bază portări ale sistemului de operare de timp real FreeRTOS. FreeRTOS este un sistem de operare open-source, ce oferă prin facilitățile sale flexibilitate crescută în proiectarea arhitecturii aplicației de timp real (e.g. prin crearea și ștergerea task-urilor, modificarea priorităților, etc.). Deoarece secvențele de cod corespunzătoare regulatele și estimatele sunt executate cu frecvențe de zeci de hertzi, era necesar ca partea de comunicație să fie rulată la frecvențe de ordinal sutelor de hertzi, ceea ce a condus la alegerea unei frecvențe de lucru a nucleului sistemului de operare de 1kHz. De

asemenea, secvențele de cod ce necesitau timpi de execuție și de reacție mult mai mici (cum ar fi preluările de date în vederea salvării și identificării) au fost executate pe baza lucrului direct cu întreruperi. Practic, implementările aplicațiilor bazate pe FreeRTOS pentru ambele microcontroller-e au dimensiuni de aproximativ 4kB, restul de memorie de pe fiecare microcontroller rămânând disponibilă pentru aplicația propriu-zisă.

I.4.1.2 Particularitățile constructive ale roboților mobili utilizați în sisteme multi-agent sau de tip pluton

Dezvoltarea de programe de timp real pentru implementarea algoritmilor de control predictivi distribuiți multiagent pe sisteme încorporate precum și testarea lor experimentală folosind un pluton de vehicule (pluton de roboți mobili) presupune cunoașterea cercetărilor făcute în acest domeniu și conștientizarea problemelor ce pot apărea precum și abordarea ingenioasă de soluții noi.

Interesante sunt cercetările făcute de Gebrewahid și Jokhio în domeniul plutoanelor de vehicule. Deși vehiculele erau reprezentate de roboți mobile, cercetările au fost realizate în direcția studiului comunicației între vehicule precum și a diferitelor scenarii ce pot apărea în cazurile reale [93]. Robotul utilizat în cadrul acestor cercetări este un robot numit PIE (engl. Platform for Intelligent Embedded Systems). Pie este dotat cu un microcontroller RISC pe 32 de biți cu nucleu ARM7 de la Atmel și cu comunicație radio implementată cu circuitul nRF24L01. Aplicația este realizată în limbajul de programare C și este bazată pe nucleul de sistem de operare TinyTimber, un nucleu de tip real pentru sistemele încorporate bazate pe evenimente. Poziționarea vehiculelor este determinată cu ajutorul unui sistem de vedere artificială dezvoltat în Matlab și este transmisă tuturor vehiculelor prin radio. Gebrewahid și Jokhio au arătat că performanțele unui pluton de vehicule depend în mare măsură de comunicația dintre vehicule precum și dintre vehicule și infrastructura de comunicație. Rețeaua de comunicație trebuie să ofere un format standard, un protocol eficient, o organizare bună și cât mai puține pierderi de date. Dat fiind că performanțele sistemului de comunicație influențează performanța algoritmilor de control implementați, este necesar ca algoritmul de control să țină cont și de caracteristicile sistemului de comunicație implementat. De asemenea, este evidențiată necesitatea existenței unui SOTR la baza rulării aplicației încorporate. Unul dintre lipsurile abordării prezentate de Gebrewahid și Jokhio constă în determinarea poziției vehiculelor prin intermediul sistemului de vedere artificial. În practică această soluție nu este aplicabilă din motive ușor de banuit.

O abordare mult mai practică din acest punct de vedere o are [94]. Platforma sa de studiu a plutoanelor de vehicule este bazată pe mașinile comercializate de către firma Carrera. Mașinile sunt dotate cu un sistem încorporat de control cu un microcontroller pe 32 de biți de la Freescale, cu comunicație radio și cu un senzor de distanță. Microcontroller-ul este MCF51JM64, rulează la o frecvență de 48MHz și dispune de 16kB de memorie Ram și 64kB de memorie pentru program. Vehiculele prezintă de asemenea senzor de viteză, implementat cu un codor optic amplasat pe osia din spate. Informațiile privind distanța față de vehiculul din față sunt furnizate de către senzorul cu infraroșu GP2D120 de la Sharp, ce poate detecta un obiect în domeniul 5cm - 50cm. Informațiile de distanță sunt preluate cu o frecvență de 800Hz. Comunicațiile între vehicule și între vehicule și stația de bază sunt realizate prin intermediul unui modul radio XBee 802.15.4. Rezultate interesante au și [95].

Fiecare vehicul din pluton conține două microcontroller-e de tip Atmega16: unul pentru senzori și unul pentru elementele de acționare (direcție și propulsie). Pe lângă senzorul de distanță este utilizat și un senzor de linie ce ajută la controlul traiectoriei vehiculului.

În experimentele realizate cu vehicule reale (nu machete) informațiile de poziționare sunt obținute cu senzori de tip LIDAR, radar și GPS. Deși senzori de tip GPS nu sunt practici pentru tema propusă, dat fiind că majoritatea experimentelor vor fi făcute în interior, testarea și utilizarea unui senzor LIDAR prezintă interes, pentru a putea determina mai bine avantajele și eventual dezavantajele utilizării unui astfel de senzor. El ar putea fi amplasat pe vehiculul cu rol de conducător din fiecare pluton și ar furniza informații detaliate privind obstacolele și arhitectura străzii din fața plutonului.

Fiecare vehicul din pluton trebuie să poată salva date într-o memorie locală, date ce pot ajuta la identificarea sistemului precum și la depanarea aplicației încorporate. Sistemul încorporat prezentat în [90] execută aceste salvări de pachete de 30 de octeți cu o frecvență de 70Hz.

I.4.2 Restricții hardware

Soluțiile pentru controlul predictiv bazat pe model (model predictive control - MPC) liniar și hibrid cu restricții, pot fi caracterizate în mod explicit sub forma unei legi de reglare după stare liniară pe porțiuni (piecewise linear - PWL). Un regulator PWL este calculat apriori utilizând programarea parametrică și implementarea exactă a regulatorului MPC explicit corespunde evaluării unei funcții PWL în unitatea de control. S-a arătat recent că o astfel de evaluare poate fi amplificată de către structurile de date de tipul arborilor binari de căutare. În [96] s-a realizat un raport referitor la proiectarea arhitecturilor hardware pentru astfel de legi de reglare PWL, și s-a arătat că soluțiile MPC explicite pot fi implementate în FPGA sau ASIC cu aproximativ 20 de mii de porți logice ducând la timpi de calcul de ordinul microsecundelor la fiecare evaluare pentru aplicații tipice. Acest lucru facilitează implementarea legilor de reglare predictive cu restricții în aplicații industriale la scară mică și în dispozitive electronice de larg consum, care sunt caracterizate printr-o dinamică rapidă sau cost redus, cum ar fi aplicații din domeniile mecatronicii, MEMS, mașini rotative, electronică de putere și acustică.

Comparativ cu legile de reglare predictive convenționale, care se bazează pe optimizare numerică în timp real, beneficiile includ verificabilitatea, complexitatea de calcul redusă, nu este nevoie de aritmetica în virgulă mobilă (fără calcule numerice recursive), și execuție deterministă. Principala limitare a abordării MPC explicite se referă la creșterea calculului computațional offline (în timpul sintezei) și a cerințelor de memorie ale unității de control, o dată cu dimensiunea și complexitatea problemei, ceea ce face utilă abordarea în special pentru probleme de dimensiuni mici. Totuși, acest lucru poate să nu fie prohibitiv în multe domenii de aplicare.

În lista de mai jos sunt prezentați parametrii esențiali care caracterizează complexitatea funcției PWL care definește legea de reglare MPC [96].

- n - Numărul de variabile de intrare în regulator ($\dim(x) = n$);
- m - Numărul de variabile de ieșire din regulator ($\dim(u) = m$);

- N - Numărul de regiuni poliedrale în partiție;
- M - Numărul de noduri al arborelui de căutare;
- D - Adâncimea arborelui de căutare (numărul maxim de noduri care trebuie să fie parcurse pentru a ajunge la un nod de tip frunză);
- H - Numărul de hyper-planuri care delimitează regiunile poliedrale.

I.5 (A1.4) Integrarea optimă a acestor modele într-un model distribuit realist pentru PV și (A1.5) Analiza modelului complet din punctul de vedere al strategiilor de control

Prin îmbinarea modelelor prezentate anterior se obține un model care surprinde atât dinamica laterală și dinamica longitudinală a autovehiculelor, cât și restricțiile introduse de comunicațiile dintre autovehiculele dintr-un pluton, precum și restricțiile hardware introduse de implementarea reguletoarelor predictive distribuite în aplicații de timp real.

Sistemele ACC sau CACC sunt capabile să mențină o viteză constantă, a vehiculului pe care sunt amplasate, raportat la viteza vehiculului care se deplasează în fața acestuia, menținând astfel și o distanță relativ constantă între cele două autovehicule. Pentru a se evita eventualele coliziuni care ar putea apărea din cauza unor erori de distanță și de viteză între vehicule se urmărește implementarea unor metode de control adecvate pentru a reduce cât mai mult posibil erorile în ceea ce privește distanța, viteza și accelerația. Se dorește de asemenea obținerea unor distanțe mai mici între membrii unui pluton pentru a fluidiza cât de mult posibil traficul. Acest lucru este posibil în cazul în care un vehicul urmăritor va cunoaște intenția vehiculului din fața sa înainte ca acesta să acționeze.

I.5.1 Metodologii de control

I.5.1.1 Reguletoare PID

În lucrarea [9] este descris un algoritm de control bazat pe o combinație între structurile de tip feed-forward și feedback deoarece un vehicul real are o dinamică neliniară. Reglarea feed-forward este alcătuită din inversul modelului vehiculului și din generatorul traiectoriilor nominale care furnizează traiectoria dorită $x^0(t)$. Pentru realizarea acestei traiectorii este necesar ca reguletorul bazat pe feedback să existe, deoarece acesta va genera mărimea de control $\delta u(t)$ pentru eliminarea erorii de traiectorie privind deviația de la traiectoria de referință. Suma mărimilor de control $u(t)$ din Fig. I.19, când reguletorul liniar este format din matricea $K(t)$, este definită prin relația:

$$u(t) = u^0(t) + \Delta u(t) = u^0(t) - K(t)\Delta x(t) = u^0(t) - K(t)[x(t) - x^0(t)]. \quad (\text{I.69})$$

Avându-se ca și punct de pornire schema bloc din Fig. I.19 se pot dezvolta algoritmii de control pentru vehicule bazați pe conceptul prezentat. Reguletorul de tip feedback, care furnizează stabilizarea obiectului în jurul traiectoriei nominale, poate fi proiectat utilizând un model liniarizat.

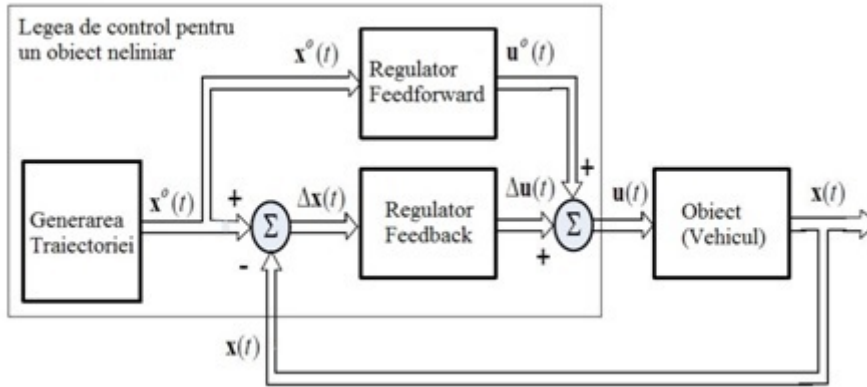


Figura I.19: Sistemul de control feed-forward și feedback pentru un model neliniar de vehicul.

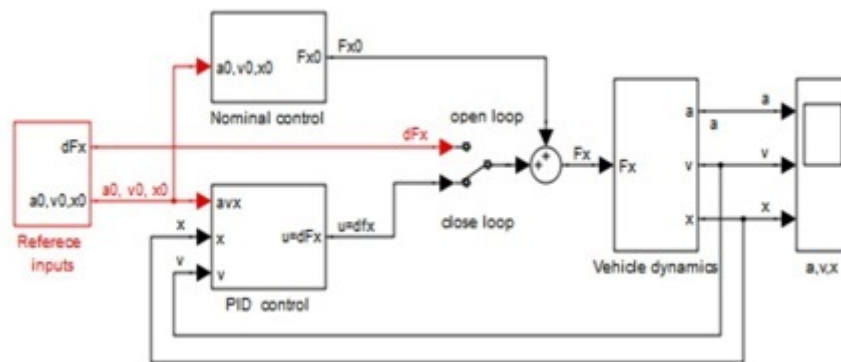


Figura I.20: Schema Simulink pentru controlul vehiculului.

Autorii acestei lucrări au ales proiectarea unui regulator PID pentru structura de tip feedback, care este obținut pe baza modelului liniar al vehiculului. Modulul de generare a intrărilor (Fig. I.20) furnizează referințe pentru accelerație a_0 , viteză v_0 , și poziție x_0 . Aceste semnale sunt transferate către regulatorul PID unde sunt procesate pe baza relației:

$$u = \Delta F_x = K_P(x_0 - x) + \frac{K_I}{s}(x_0 - x) + K_D(v_0 - v), \quad (I.70)$$

unde K_P , K_I și K_D sunt parametrii proporțional, integrator și derivativ ai regulatorului, iar a , v și x sunt accelerația, viteza și poziția vehiculului.

Modulul Nominal control și modulul Vehicle dynamics din Fig. I.20 se bazează pe ecuațiile următoare:

$$F_x^0 = mg \sin \theta^0 + f_r mg \cos \theta^0 + \frac{1}{2} C_{air} (u^0 + u_w)^2, \quad (I.71)$$

$$m \dot{u} = F_x - mg \sin \theta - f_r mg \cos \theta - \frac{1}{2} C_{air} (u + u_w)^2, \quad (I.72)$$

unde F_x este forța de tracțiune, u_w este viteza vântului, f_r este coeficientul rezistenței la rulare, iar $C_{air} = \rho A_r C_d$ este o constantă.

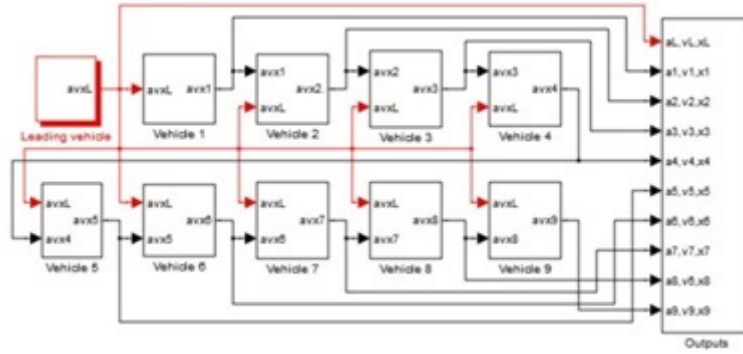


Figura I.21: Modelul Simulink al unui pluton cu 10 vehicule.

Deplasarea vehiculelor se face în sistemul de coordonate absolut $G(O; x_0, y_0)$ care este fixat pe drum cu originea în punctul de start, O . Pozițiile x_i , vitezele $v_i = \dot{x}_i$, și accelerațiile $a_i = \dot{v}_i$, $i = L, 1, 2, 3, 4$, măsurate cu respectarea sistemului $G(O; x_0, y_0)$, sunt valori absolute. Sistemul de coordonate $L(L; x_L, y_L)$ este fixat pe vehiculul lider cu originea în centrul masei lui. Poziția relativă, viteza și accelerația vehiculelor cu referire la sistemul $L(L; x_L, y_L)$ sunt notate astfel: $l_i = x_L - x_i$, $v_{ri} = v_L - v_i$, $a_{ri} = a_L - a_i$, $i = 1, 2, 3, 4$. Distanțele dintre vehicule sunt notate cu $dx_i = x_{i-1} - x_i$, $i = L, 1, 2, 3, 4$, iar vitezele și accelerațiile relative ale vehiculelor au expresiile: $dv_i = v_{i-1} - v_i = \dot{x}_{i-1} - \dot{x}_i$, respectiv $da_i = a_{i-1} - a_i = \ddot{x}_{i-1} - \ddot{x}_i$.

În Fig. I.21 este reprezentată o schemă a unui pluton în care toate vehiculele au aceleași caracteristici și obțin informații despre accelerație, viteză și poziție de la vehiculele precedente și de la vehiculul lider.

Utilizând modelul vehiculului (I.72) și considerând că $\theta = 0$ și $V_w = 0$, se poate determina accelerația vehiculului cu această formă:

$$\begin{cases} \dot{u} = a = \frac{1}{m} (F_x - f_r mg - \frac{1}{2} C_{air} u^2) \\ F_x = \Delta F_x + F_{x0} \end{cases} \quad (I.73)$$

Forța de control ΔF_x este determinată de regulatorul PID. Prin înlocuirea ecuației (I.70) în relația (I.73) se determină accelerația celui de-al i -lea vehicul:

$$a_i = \frac{1}{m} [K_{Pi} (x_{i-1} - x_i - hd_i) + \frac{K_{Ii}}{s} (x_{i-1} - x_i - hd_i) + K_{Di} (v_{i-1} - v_i) + F_{x0} - f_r mg - \frac{1}{2} C_{air} u_i^2], \quad (I.74)$$

unde hd_i este distanța constantă dintre vehiculele $i - 1$ și i . Prin derivarea relației (I.73) și folosind relațiile:

$$\dot{x}_i = v_i, \quad (I.75)$$

$$\dot{v}_i = a_i, \quad (I.76)$$

se obține gradul de mișcare sacadată care acționează asupra celui de-al i -lea vehicul:

$$\dot{a}_i = \frac{1}{m} [K_{Ii} (x_{i-1} - x_i - hd_i) + K_{Pi} (v_{i-1} - v_i) + K_{Di} (a_{i-1} - a_i) - C_{air} u^0 a_i]. \quad (I.77)$$

Ecuțiile (I.75) și (I.76) reprezintă modelul liniar în spațiul stărilor al vehiculului i din pluton și pot fi utilizate pentru obținerea modelului unui șir de vehicule.

Se consideră modelul unui șir de vehicule cu trei membri: vehiculul lider și două vehicule următoare. Mărimile de ieșire ale liderului generează variabilele de intrare, x_L , v_L și a_L , pentru primul vehicul din șir. Celelalte două vehicule sunt descrise cu ecuațiile:

- primul vehicul:

$$\begin{aligned} \dot{x}_1 &= v_1 \\ \dot{v}_1 &= a_1 \\ \dot{a}_1 &= \frac{1}{m} [K_{I1}(x_L - x_1 - hd_1) + K_{P1}(v_L - v_1) + K_{D1}(a_L - a_1) - C_{air}u^0 a_1] \end{aligned} \quad (I.78)$$

- al doilea vehicul:

$$\begin{aligned} \dot{x}_2 &= v_2 \\ \dot{v}_2 &= a_2 \\ \dot{a}_2 &= \frac{1}{m} [K_{I2}(x_1 - x_2 - hd_2) + K_{P2}(v_1 - v_2) + K_{D2}(a_1 - a_2) - C_{air}u^0 a_2] \end{aligned} \quad (I.79)$$

Se poate forma vectorul de stări $x = [x_1 \ dx_2 \ v_1 \ v_2 \ a_1 \ a_2]^T$ în care starea dx_2 este distanța dintre primul și al doilea vehicul, iar $\dot{x}_2 = \dot{x}_1 - \dot{x}_2 = v_1 - v_2 = dv_2$.

Exprimarea ecuației șirului de vehicule în reprezentarea vector-matrice este de forma:

$$\begin{aligned} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{v}_1 \\ \dot{v}_2 \\ \dot{a}_1 \\ \dot{a}_2 \end{bmatrix} &= \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ -\frac{K_{I1}}{m} & 0 & -\frac{K_{P1}}{m} & 0 & \frac{-K_{D1}-C_{air}u^0}{m} & 0 \\ 0 & \frac{K_{I2}}{m} & \frac{K_{P2}}{m} & -\frac{K_{P2}}{m} & \frac{K_{D2}}{m} & \frac{-K_{D2}-C_{air}u^0}{m} \end{bmatrix} \begin{bmatrix} x_1 \\ dx_2 \\ v_1 \\ v_2 \\ a_1 \\ a_2 \end{bmatrix} + \\ &+ \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \\ \frac{K_{I1}}{m} & \frac{K_{P1}}{m} & \frac{K_{D1}}{m} \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_L \\ v_L \\ a_L \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ -\frac{K_{I1}}{m} & 0 \\ 0 & -\frac{K_{I2}}{m} \end{bmatrix} \begin{bmatrix} hd_1 \\ hd_2 \end{bmatrix} \end{aligned} \quad (I.80)$$

Se poate forma vectorul ieșirilor $y = [dx_2 \ v_1 \ v_2]^T$ cu expresia:

$$y = \begin{bmatrix} dx_2 \\ v_1 \\ v_2 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \end{bmatrix} x + \begin{bmatrix} 0 & 0 & 0 \\ 0 & 0 & 0 \\ 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} x_L \\ v_L \\ a_L \end{bmatrix} \quad (I.81)$$

Pentru un pluton de vehicule se definește și stabilitatea șirului, pe lângă stabilitatea individuală a fiecărui vehicul. Dacă vehiculul precedent accelerează sau decelerează, eroarea de distanță poate deveni mai mare decât zero. Trebuie să se acționeze în vederea atenuării propagării erorii de distanță de-a lungul șirului până la finalul acestuia.

Transformând ecuația (I.75) în domeniul Laplace și utilizând funcția de transfer $G_i = v_i/v_{i-1}$ se poate determina eroarea de distanță dintre cel de-al i -lea și al $(i-1)$ -lea vehicul:

$$\begin{aligned} x_i &= \frac{1}{s}v_i \\ v_i &= G_i(s)v_{i-1} \quad , \\ \varepsilon_i &= x_{i-1} - x_i - D_i \end{aligned} \quad (I.82)$$

unde $D_i = h_i v_i$ reprezintă distanța dorită pentru vehiculul i , h_i este tactica în ceea ce privește progresul în timp adoptată pentru toate vehiculele. Pornind de la eroarea de distanță a celui de-al i -lea vehicul până la eroarea celui de-al $(i+k)$ -lea vehicul poate fi definită funcția de transfer:

$$G_{i,k}(s) = \frac{\varepsilon_{i+k}}{\varepsilon_i} = G_i G_{i+1} \dots G_{i+k-1} \frac{1 - G_{i+k} - s h_{i+k} G_{i+k}}{1 - G_i - s h_i G_i}. \quad (I.83)$$

Pentru ca stabilitatea șirului să fie satisfăcută trebuie să fie îndeplinite următoarele condiții:

$$\|\varepsilon_{i+k}\| \leq \|\varepsilon_i\| \text{ sau } \|G_{i,k}(s)\|_{\infty} \leq 1. \quad (I.84)$$

I.5.1.2 Metoda Sliding Mode Control (SMC)

În articolul [97] este descris faptul că sistemul de control longitudinal dezvoltat pentru deplasarea în pluton a vehiculelor constă dintr-un regulator de pe nivel superior (upper-level controller - ULC) și un regulator de pe nivel inferior (lower-level controller - LLC). Cel de nivel superior determină accelerația dorită pentru fiecare vehicul din pluton. Regulatorul de pe nivelul inferior are ca scop determinarea comenzilor de accelerare sau frânare pentru a urmări accelerația dorită.

Regulatorul de pe nivelul superior

Regulatorul de pe nivelul superior (ULC) are rolul de a determina accelerația dorită pentru fiecare vehicul astfel încât:

1. să mențină o distanță constantă mică între mașini;
2. să asigure stabilitate plutonului.

Modelul părții fixate considerat pentru proiectarea regulatorului în cauză este:

$$\ddot{x}_i = u \quad (I.85)$$

unde indicele i reprezintă numărul vehiculului din pluton. Intrarea de control pentru acest sistem este accelerația vehiculului. Din cauza faptului că banda asociată regulatorului LLC este finită, se așteaptă ca fiecare vehicul să urmărească în mod imperfect accelerația dorită. Asigurarea performanței regulatorului ULC se va concentra în jurul îndeplinirii robuste a obiectivelor 1 și 2 în prezența unei întârzieri de gradul întâi în performanța LLC:

$$\ddot{x}_i = \frac{1}{\tau s + 1} \ddot{x}_{i,des} = \frac{1}{\tau s + 1} u_i \quad (I.86)$$

Eroarea de distanță pentru al i -lea vehicul este definită ca $\varepsilon_i = x_i - x_{i-1} + L$ unde L este distanța dorită. Luând în considerare eroarea de distanță cele două obiective mai sus enunțate pentru ULC pot fi rescrise astfel:

1.

$$\varepsilon_{i-1} \rightarrow 0 \Rightarrow \varepsilon_i \rightarrow 0, \quad (\text{I.87})$$

2.

$$\left\| \hat{H}(s) \right\|_{\infty} \leq 1, \quad (\text{I.88})$$

unde $\hat{H}(s)$ este funcția de transfer care face referire la eroarea de distanță dintre două vehicule consecutive din pluton:

$$\hat{H}(s) = \frac{\varepsilon_i}{\varepsilon_{i-1}}. \quad (\text{I.89})$$

Stabilitatea șirului de vehicule (obiectivul 2= se referă la o proprietate în care erorile de distanță, detectate la un moment dat de-a lungul șirului, nu se amplifică în timpul propagării acestora până la finalul plutonului. Pe lângă condițiile (I.87) și (I.88) mai este de dorit ca funcția răspuns la impuls $h(t)$ corespunzătoare $\hat{H}(s)$ să nu își schimbe semnul.

Algoritmii pentru controlul longitudinal care garantează stabilitatea șirului de vehicule includ algoritmi autonomi, semi-autonomi și bazați pe comunicații radio.

Pentru controlul longitudinal se folosește metoda sliding mode control (SMC). Suprafața de alunecare (sliding surface) definită de această metodă are forma:

$$S_i = \dot{\varepsilon}_i + \frac{\omega_n}{\xi + \sqrt{\xi^2 - 1}} \frac{1}{1 - C_1} \varepsilon_i + \frac{C_1}{1 - C_1} (v_i - v_l), \quad (\text{I.90})$$

unde ω_n este banda de funcționare a ULC, C_1 este ponderea relativă pentru compararea semnalului de tip feedback a vehiculului lider cu semnalul corespunzător vehiculului din fața sa, iar v_l este viteza liderului.

Considerând

$$\dot{S}_i = -\lambda S_i \text{cu } \lambda = \omega_n \left(\xi + \sqrt{\xi^2 - 1} \right) \quad (\text{I.91})$$

se determină accelerația dorită a vehiculului cu forma:

$$\begin{aligned} \ddot{x}_{i.des} = & (1 - C_1) \ddot{x}_{i-1} + C_1 \ddot{x}_l - \left(2\xi - C_1 \left(\xi + \sqrt{\xi^2 - 1} \right) \right) \omega_n \dot{\varepsilon}_i - \\ & - \left(\xi + \sqrt{\xi^2 - 1} \right) \omega_n C_1 (v_i - v_l) - \omega_n^2 \varepsilon_i. \end{aligned} \quad (\text{I.92})$$

Parametrii de control care trebuie acordați sunt C_1 , ζ și ω_n . Factorul C_1 poate lua valori $0 \leq C_1 < 1$ și poate fi privit ca o pondere între a vitezei și accelerației vehiculului lider. Termenul ζ este raportul de amortizare și poate fi setat cu valoarea 1 pentru amortizări critice. ω_n este banda de funcționare a regulatorului.

Ecuția (I.91) asigură ca suprafața de alunecare să atingă valoarea zero. Dacă toate vehiculele din pluton au implementată această lege de control este posibil ca acestea să urmărească vehiculul precedent cu o distanță constantă între ele și se asigură că plutonul este stabil, în sensul că erorile de distanță nu se propagă de-a lungul șirului. Pentru a avea acces la toate semnalele necesare se folosește un sistem de comunicații radio wireless. Fiecare autovehicul obține informații despre viteză și accelerație de la vehiculul lider și de la cel precedent.

Setând pentru un pluton cu două vehicule se obține un sistem clasic de ordinul doi:

$$\ddot{x}_{i.des} = \ddot{x}_{i-1} - 2\xi\omega_n\dot{\varepsilon}_i - \omega_n^2\varepsilon_i. \quad (I.93)$$

Regulatorul de pe nivelul inferior

În regulatorul de pe nivelul inferior (LLC) intrările de acționare pentru accelerare și frânare sunt determinate astfel încât să urmărească accelerația dorită descrisă de relația (I.92). Pentru implementarea regulatorului LLC se consideră modelul simplificat al dinamicii vehiculului cu expresia:

$$\dot{x}_i = v_i = (Rh\omega_e)_i, \quad (I.94)$$

unde R este raportul de transmisie, h reprezintă raza anvelopei, iar ω_e este viteza unghiulară a motorului. Acest model simplificat se bazează pe presupunerile că convertorul de cuplu din vehicul este blocat și că alunecarea dintre anvelopă și drum este egală cu zero. Aceste aproximări fac posibilă transpunerea vitezei vehiculului direct în viteza motorului.

Dinamica privitoare la viteza motorului ω_e și la pseudointrările cuplul de ardere net T_{net} și cuplul de frânare T_{br} poate fi modelată prin:

$$\dot{\omega}_e = \frac{T_{net} - c_a R^3 h^3 \omega_e^2 - R(hF_f + T_{br})}{J_e}, \quad (I.95)$$

unde $J_e = I_e + (mh^2 + I_\omega) R^2$ este inerția efectivă în ceea ce privește motorul, c_a este coeficientul de rezistență aerodinamică, iar F_f reprezintă rezistența la rulare a anvelopelor.

$T_{net}(\omega_e, m_a)$ este o funcție neliniară ce depinde de turație motorului și de masa aerului din galeria de admisie. Dinamica care leagă m_a de unghiul de accelerare α poate fi modelată astfel:

$$\dot{m}_a = \dot{m}_{ai} - \dot{m}_{ao}, \quad (I.96)$$

unde \dot{m}_{ai} și \dot{m}_{ao} sunt debitul de aer la intrarea în galeria de admisie și la ieșirea din aceasta, și

$$\dot{m}_{ai} = MAXTC(\alpha) PRI(m_a), \quad (I.97)$$

în care MAX este o constantă dependentă de corpul clapetei de accelerație, $TC(\alpha)$ este o funcție inversabilă neliniară al unghiului de accelerație și PRI reprezintă funcția de influență a presiunii care descrie gradul de asfixiere a fluxului de curgere care apare în valva de accelerație. \dot{m}_{ao} este masa debitului de aer din camera de ardere. Pentru a se descrie fluxul de aer din galeria de admisie se utilizează legea gazelor ideale:

$$P_m V_m = m_a R_g T. \quad (I.98)$$

Proiectarea legii de control pentru regulatorul LLC se bazează pe o modificare a tehnicii de control SMC standard. Dacă cuplul de ardere net este ales cu expresia:

$$(T_{net})_i = \frac{J_e}{Rh} \ddot{x}_{i.des} + [c_a R^3 h^3 \omega_e^2 + R(hF_f + T_{br})]_i \quad (I.99)$$

atunci din relația (I.95) se deduce că accelerația vehiculului este egală cu accelerația dorită determinată de regulatorul ULC: $\ddot{x}_i = \ddot{x}_{i.des}$.

Cunoscând expresia cuplului de ardere din relația (I.99) unghiul de accelerare necesar pentru a obține acest cuplu se calculează după următoarea procedură:

- se măsoară presiunea aerului în galeria de admisie P_m și temperatura T și apoi se calculează m_a utilizând legea gazelor ideale (I.98);
- funcția $T_{net}(\omega_e, m_a)$ este inversată pentru a se obține $m_{a.des}$;
- apoi este utilizat un regulator SMC pentru a calcula unghiul α necesar pentru ca mărimea $m_{a.des}$ să urmărească m_a .

Se definește suprafața de alunecare de forma:

$$s_2 = m_a - m_{a.des}. \quad (I.100)$$

Prin stabilirea $\dot{s}_2 = -\eta_2 s_2$, se obține:

$$\text{MAXTC}(\alpha) \text{PRI}(m_a) = \dot{m}_{ao} + \dot{m}_{a.des} - \eta_2 s_2, \quad (I.101)$$

unde η_2 este un factor de ponderare pentru metoda sliding mode control.

Atât timp cât funcția $TC(\alpha)$ este inversabilă poate fi calculat unghiul de accelerare din relația (I.101). Dacă cuplul net de ardere este negativ, se folosește dispozitivul de frânare pentru a genera cuplul dorit.

I.5.1.3 Metoda Model Predictive Control (MPC)

Autorii lucrării [62] au propus pentru controlul unui pluton de autovehicule proiectarea unui regulator bazat pe predicție (Model Predictive Control MPC) deoarece capacitatea de calcul în creștere oferă soluții adecvate pentru sistemele discrete cu frecvențe mari ale perioadei de eșantionare. Bazându-se pe un model matematic al sistemului, este rezolvată o problemă de optimizare asupra unui orizont de predicție finit având lungimea $N_p T_s$ cu o perioadă de eșantionare constantă T_s .

Variabilele de intrare pot fi diferite de zero peste un orizont de predicție cu lungimea $N_c \cdot T_s$ ($N_c < N_p$). De obicei doar prima valoare a secvenței de intrare prezise este aplicată sistemului.

Se utilizează următoarea notație: $x_{t+k|t}$ cu $t > 0, k \geq 1$ este a k -a valoare de stare prezisă, pe baza valorii de stare inițială $x_{t|t} := x(t)$ și a secvenței de intrare u_t, \dots, u_{t+k-1} . Secvența de stări prezise la momentul de timp inițial t și având secvența intrărilor predictate $U := \{u_t, \dots, u_{t+N_c-1}\}$ este $X := \{x_{t+1|t}, \dots, x_{t+N_p|t}\}$.

Evaluând sistemul de matrici (I.54) se poate determina că toate valorile proprii λ conduc la $\lambda = 1$. Analiza stabilității necesită ca toate valorile proprii λ ale sistemului A să fie distincte cu $|\lambda| < 1$. Acest lucru poate fi asigurat prin aplicarea unui regulator de stare liniar cu un factor K de forma:

$$u(k) := -Kx(k) + \tilde{u}(k) \quad (I.102)$$

cu o nouă variabilă de intrare $\tilde{u}(k)$ pentru $k \geq 0$. Astfel se obține un nou sistem de ecuații:

$$\begin{cases} x(k+1) = Ax(k) + Bu(k) = \underbrace{(A - BK)}_{=: \tilde{A}} x(k) + B\tilde{u}(k) \\ y(k) = x(k) \end{cases} \quad (I.103)$$

pentru toți $k \geq 0$.

Atât timp cât modelul este complet controlabil, se poate alege K , astfel încât toate valorile proprii λ ale \tilde{A} sunt distincte și $0.9 \leq \lambda < 1$. Aceasta poate fi calculată prin utilizarea funcției `place` din MATLAB.

Se poate astfel formula problema de optimizare P cu $N_c < N_p$ cum se poate observa în relația (I.104). În continuare se consideră modelul (I.103) cu matricile din relațiile (I.54)-(I.57), dar se va omite notația pentru simplitate. Se vor considera restricțiile $x_{\min}x_{\max}$, matricile de ponderare nesingulare $Q \in \mathbb{R}^{4n \times 4n}$, $R \in \mathbb{R}^{n \times n}$ și $P \in \mathbb{R}^{4n \times 4n}$ matricea de rang complet.

$$P : \begin{cases} \min_U \left\{ \|Px_{t+N_p|t}\|_1 + \sum_{k=1}^{N_p-1} \|Qx_{t+k|t}\|_1 + \sum_{k=0}^{N_c-1} \|Ru_{t+k}\|_1 \right\} \\ s.t. \\ x_{t|t} = x(t) \\ x_{t+k+1|t} = Ax_{t+k|t} + Bu_{t+k}; \forall k \geq 0, \\ u_{t+k} = 0; N_c \leq k \leq N_p - 1, \\ x_{\min} \leq x_{t+k|t} \leq x_{\max}; k = 1, \dots, N_p \end{cases} \quad (\text{I.104})$$

Datorită faptului că în funcția de cost se folosește norma de gradul 1 aceasta prezintă două avantaje majore:

Un vehicul din pluton reacționează doar la vehiculele precedente și nu și la cele care urmează după acesta în șir.

O problemă de optimizare liniară necesită un timp de calcul mai scurt.

În [98] autorii prezintă stabilitatea asimptotică a problemei P prin alegerea adecvată a matricii de pondere P . Pentru aceasta trebuie să se garanteze fezabilitatea sistemului prin alegerea unei valori a N_c suficient de mare. Cu referire la [99] se poate determina seria admisibilă cu următoarele condiții îndeplinite:

1. A este asimptotic stabilă (toate valorile proprii λ ale A sunt distincte și $|\lambda_i(A)| < 1$ pentru $i = 1, \dots, 4n$),
2. \mathbb{X} este mărginit,
3. $0 \in \text{int}\mathbb{X}$.

I.5.2 Modele de vehicule utilizate în controlul predictiv distribuit

Controlul predictiv bazat pe model (CPM) este o tehnică de control optimală care folosește un model dinamic pentru a prezice evoluția sistemului pe durata unui orizont de predicție viitor și folosește această predicție pentru a produce la momentul curent, cea mai bună strategie de comandă optimizând o funcție de cost specifică [100]. Din traiectoria de comandă, doar primul element se folosește ca semnal de intrare către proces. Această tehnică poartă numele de principiul orizontului alunecător [101], deoarece orizontul de predicție alunecă pe axa temporală și întreaga procedură de calcul se repetă la fiecare perioadă de eșantionare.

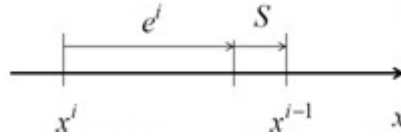


Figura I.22: Modelul local al agentului i , cu $d^i = e^i + S$.

Controlul predictiv distribuit presupune partiționarea unui sistem de mari dimensiuni în subsisteme cuplate între ele prin dinamică (intrări sau stări) [30]. Astfel, fiecare subsistem este controlat de un regulator propriu care optimizează o funcție de cost locală considerând în problema de optimizare și informații primite de la vecini (i.e. totalitatea subsistemelor cuplate între ele).

În acest capitol este prezentată analiza literaturii de specialitate, evidențind modelele disponibile pentru plutoanele de autovehicule, în scopul implementării controlului predictiv distribuit. Datorită particularităților acestor sisteme și anume structura în lanța subsistemelor componente, acestea pot fi reglate prin tehnici de control predictiv distribuit. Una dintre cele mai uzuale aplicații o reprezintă controlul plutonului de autovehicule cu păstrarea unei distanțe constante între subsisteme.

Schemele de control predictiv distribuit folosite în cadrul acestor sisteme pot fi clasificate în funcție de tipul interacțiunii dintre subsisteme. Astfel, cuplajul dintre subsisteme poate fi exprimat prin introducerea unor termeni specifici în funcția cost [102] sau prin introducerea unor restricții specifice asupra stărilor sistemului [103].

În [104] este prezentată o schemă de control predictiv distribuit robust aplicată plutoanelor de autovehicule în care intrarea subsistemului anterior este considerată perturbare deoarece valoarea acesteia nu este cunoscută local în cadrul fiecărui subsistem. În acest caz, problema de control predictiv este aplicată asupra sistemului nominal (fără perturbații), introducând o restricție de robustețe asupra traiectoriei predictate a stării.

În aceasta lucrare, s-a abordat problema controlului unui pluton de autovehicule păstrând constantă distanța dintre ele. Fie N_A vehicule ce se deplasează urmărind o referință. În aceste condiții, este suficientă modelarea mișcării fiecărui subsistem doar pe o singură direcție, scopul fiind ca toate vehiculele să urmărească vehiculul predecesor menținând o distanță constantă între ele notată S . Fiecare agent (i.e. regulator alocat subsistemului, care elaborează legea de control folosind informații primite de la regulatoarele subsistemelor cu care interacționează) poate măsura distanța D^i și viteza relativă \dot{d}^i a predecesorului său. Fiecare vehicul i este modelat ca dublu integrator unde mărimea de control este accelerația și stările sunt poziția p^i și viteza v^i . Pe baza stării locale x^i poate fi definit eroarea e^i raportată la distanța specificată și derivate în timp a erorii, i.e., $x^i = [e^i \dot{e}^i]^T$ (vezi Fig. I.22).

Dacă sistemul în timp continuu este discretizat cu o perioadă de eșantionare T , rezultă modelul în timp discret de forma:

$$x_{k+1}^i = \begin{bmatrix} 1 & T \\ 0 & 1 \end{bmatrix} x_k^i + \begin{bmatrix} -T^2/2 \\ -T \end{bmatrix} u_k^i + \begin{bmatrix} T^2/2 \\ T \end{bmatrix} u_k^{i-1} \quad (\text{I.105})$$

Obiectivul legii de control este aducerea ambelor stări în zero, ceea ce înseamnă că vehiculul se deplasează cu aceeași viteză ca și predecesorul său păstrând distanța constantă S .

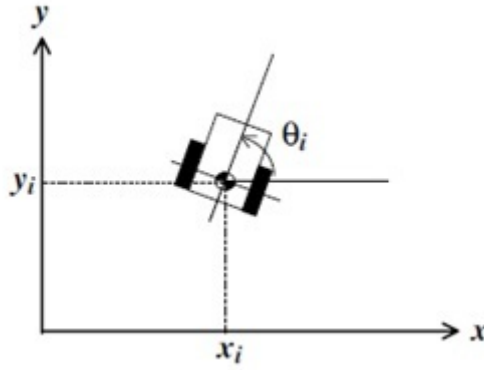


Figura I.23: Vehicul de tip unicyclu.

Acest model simplificat al unui vehicul, sub forma unui dublu integrator este regăsit și în [105].

În [106] este prezentat un algoritm de control predictiv distribuit cu restricții impuse pentru a evita coliziunile între vehicule. În această lucrare, fiecare vehicul este modelat ca un unicyclu (i.e. un vehicul care se poate deplasa simultan în două dimensiuni: mișcare de rotație și de translație). Algoritmul de control predictiv distribuit este folosit pentru a preveni creșterea complexității problemei de optimizare, odată cu mărirea dimensiunii formației de vehicule. Astfel, fiecare vehicul rezolvă o problemă de optimizare locală utilizând informațiile primite de la celelalte vehicule în scopul evitării coliziunilor exprimate sub formă de restricții.

Fie un set de n vehicule de tip unicyclu indexate cu $i = 1, \dots, n$:

$$\dot{x}_i = v_i \cos \theta_i, \dot{y}_i = v_i \sin \theta_i, \dot{\theta}_i = \omega_i \quad (\text{I.106})$$

unde v_i și ω_i sunt viteza liniară și unghiulară ale vehiculului i și (x_i, y_i, θ_i) reprezintă coordonatele măsurabile relativ la un sistem de coordonate global (vezi Fig. I.23).

Vehiculul leader (i.e. primul vehicul din formație care nu are alt vehicul predecesor) este descris de următoarele ecuații:

$$\dot{x}_r = v_r \cos \theta_r, \dot{y}_r = v_r \sin \theta_r, \dot{\theta}_r = \omega_r \quad (\text{I.107})$$

unde v_r și ω_r sunt vitezele liniară și unghiulară și (x_r, y_r, θ_r) reprezintă coordonatele măsurabile relativ la un sistem de coordonate global.

Poziția vehiculului i din (2) este exprimată în sistemul de coordonate al leader-ului (3) printr-un vector constant (r_i, l_i) (vezi Fig. I.24). Astfel, traiectoria de referință pentru vehiculului i poate fi exprimată în sistemul de coordonate global ca:

$$z_i^d := \begin{bmatrix} x_r + r_i \sin \theta_r + l_i \cos \theta_r \\ y_r - r_i \cos \theta_r + l_i \sin \theta_r \end{bmatrix} \quad (\text{I.108})$$

În (I.106), vehiculului i ($= 1, \dots, n$) este numit vehicul follower i (adică un vehicul care are un alt vehicul ca predecesor).

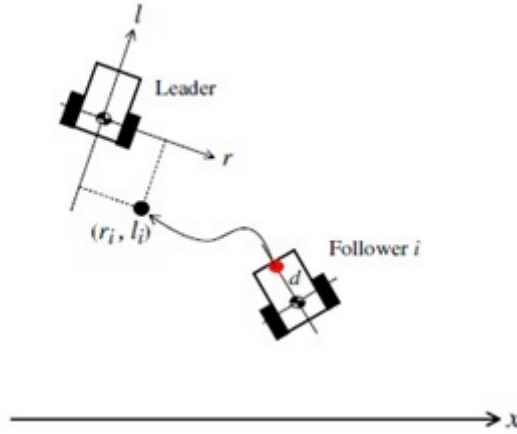


Figura I.24: Vehicul leader și vehicul follower.

Scopul este de a controla poziția fiecărui vehicul follower la o distanță constantă d definită ca:

$$z_i := \begin{bmatrix} x_{vi} \\ y_{vi} \end{bmatrix} = \begin{bmatrix} x_i + d \cos \theta_i \\ y_i + d \sin \theta_i \end{bmatrix} \quad (\text{I.109})$$

raportată la traiectoria de referință z_i^d definită în (I.108), fără coliziuni.

Controlul unei formații de vehicule de tip uniciclu a fost abordată și în [107].

În [108] este elaborată o strategie de control predictiv pentru a ghida un grup de vehicule follower într-o anumită regiune cu ajutorul mai multor vehicule leader. Se consideră un grup de sisteme identice modelate folosind dinamica sistemelor liniare în timp continuu:

$$\begin{aligned} \dot{x}_i &= Ax_i + Bu_i \\ y_i &= Cx_i, i = 1, \dots, N \end{aligned} \quad (\text{I.110})$$

unde $x_i \in \mathbb{R}^n$, $u_i \in \mathbb{R}^p$ și $y_i \in \mathbb{R}^q$ sunt stările, mărimea de comandă și ieșirea măsurată a subsistemului i și matricile A, B, C au dimensiuni adecvate. Strategia de control a fost testată folosind în (I.110) modelul vehiculelor din platforma de testare multi-vehicul wireless elaborată de Caltech [109]:

$$\begin{aligned} x_i &= \begin{bmatrix} x_{i1} & x_{i2} & x_{i3} & \dot{x}_{i1} & \dot{x}_{i2} & \dot{x}_{i3} \end{bmatrix}^T \\ A &= \begin{bmatrix} 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & -2.003 & -2.003 & 0 & 0 \\ 0 & 0 & 0.2003 & 0 & -0.2003 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1.6129 \end{bmatrix} \\ B &= \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0.9441 & 0.9441 \\ 0.9441 & 0.9441 \\ -28.7097 & 28.7097 \end{bmatrix} \end{aligned} \quad (\text{I.111})$$

unde x_{i1} și x_{i2} sunt pozițiile vehiculului i în coordonatele x și y respective, iar x_{i3} este orientarea vehiculului i .

În [110] este prezentat un algoritm de control predictiv distribuit robust aplicat unui grup de subsisteme neliniare cu restricții pe intrări și perturbații externe. Fiecare subsistem poate comunica cu vecinii săi (i.e. subsistemele cu care interacționează), având loc un schimb al traiectoriilor predictate de stare. Rezultatul a fost obținut modelând fiecare subsistem sub forma unui sistem masă-amortizor-arc:

$$\begin{cases} \dot{x}_{i1}(t) = x_{i2}(t) \\ \dot{x}_{i2}(t) = -\frac{k_i}{M_i}e^{-x_{i1}(t)}x_{i1}(t) - \frac{h_i}{M_i}x_{i2}(t) + \frac{u_i(t)}{M_i} + \frac{\omega_i(t)}{M_i} \end{cases} \quad (\text{I.112})$$

unde x_{i1} este deplasarea masei, x_{i2} este viteza masei, k_i este factorul liniar al arcului, h_i este factorul de amortizare, M_i este masa, $u_i(t)$ este forța de control și ω_i este perturbația externă.

Valorile parametrilor sunt: $h_i = 1.1 \text{Ns/s}$; $M_i = 1.5 \text{kg}$; $k_i = 0.25 \text{N/m}$. Forța de control a fost limitată astfel $u_i(t) \in [-2, 2]$.

În [111] este prezentată o schemă cu două nivele ierarhice pentru controlul unei formații de vehicule. În primul nivel ierarhic este utilizat un algoritm de control predictiv pentru elaborarea traiectoriei formației (i.e. calculul mărimilor de comandă), iar cel de-al doilea nivel conține regulatorul pentru controlul formației de vehicule (corectarea micilor deviații de la traiectoria de referință și menținerea distanței între vehicule). Algoritmul a fost testat pe o formație de vehicule acționate diferențial (i.e. vehicul cu două roți care are un element de execuție corespunzător fiecărei roți). Fiecare vehicul (vezi Fig. I.25) are următorul model cinematic:

$$\begin{cases} \dot{x}(t) = (u_1(t) - u_2(t)) \cos \theta(t) \\ \dot{y}(t) = (u_1(t) - u_2(t)) \sin \theta(t) \\ \dot{\theta}(t) = (u_1(t) + u_2(t)) \end{cases} \quad (\text{I.113})$$

unde $\theta(t) \in [-\pi, \pi]$ și mărimile de comandă $u_1(t), u_2(t) \in [-1, 1]$.

Coordonatele (x, y) sunt poziția în plan a punctului situat la jumătatea axului ce conectează roțile și θ este unghiul raportat la axa x . Mărimile de comandă u_1, u_2 sunt vitezele unghiulare ale roții drepte respectiv stângi. Dacă se aplică aceeași viteză pe ambele roți, atunci vehiculul se deplasează în linie dreaptă (cu viteza maximă de înaintare $u_1 = u_2 = u_{\max} = 1$). Vehiculul se întoarce dacă $u_1 \neq u_2$ (e.g. dacă $u_1 = -u_2 = 1$ atunci vehiculul se întoarce în sensul invers acelor de ceasornic în jurul punctului median al axului).

Modelul liniarizat al vehiculului se determină plecând de la presupunerea că formația se deplasează cu o viteză liniară nominală $v_n = (u_{1n} + u_{2n})/2$ unde u_{1n}, u_{2n} sunt vitezele nominale ale roților. Fie, v_l o viteză liniară adăugată la viteza liniară nominală și v_ω o viteză unghiulară. Totodată, se presupune că unghiul θ este mic, astfel încât $\cos \theta = 1$ și $\sin \theta = 0$. Modelul simplificat obținut are următoarea formă:

$$\begin{cases} \dot{x}(t) = (v_n(t) + v_l(t)) \\ \dot{y}(t) = (v_n(t) + v_l(t)) \theta(t) \\ \dot{\theta}(t) = v_\omega(t) \end{cases} \quad (\text{I.114})$$

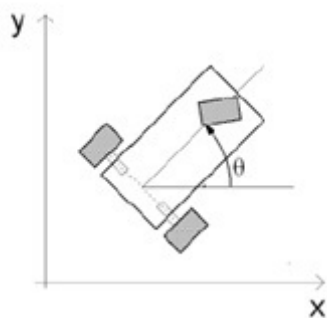


Figura I.25: Vehicul acționat diferențial.

Modelul liniarizat este obținut având axa z_1 aliniată cu viteza vehiculului leader:

$$\begin{bmatrix} z_1(t+h) \\ z_2(t+h) \\ z_3(t+h) \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & v_n h \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} z_1(t) \\ z_2(t) \\ z_3(t) \end{bmatrix} + \begin{bmatrix} h & 0 \\ 0 & 0 \\ 0 & h \end{bmatrix} \begin{bmatrix} v_l(t) \\ v_\omega(t) \end{bmatrix} \quad (\text{I.115})$$

unde h este perioada de esantionare.

Această modelare a vehiculelor din pluton de tip diferențiat se regăsește și în [112].

Datorită complexității algoritmului de control predictiv distribuit, cele mai uzuale modele individuale de vehicule în formație pluton regăsite în literatura de specialitate sunt simplificate de tip dublu integrator [104, 105], vehicul uniciclu [106], vehicul acționat diferențial [111, 112] sau folosind un model mai complex elaborat de Caltech [108, 109].

I.6 Concluzii

Prezentul raport a avut următoarea structură. În capitolul 1 au fost prezentate obiectivele generale ale proiectului, iar în Capitolul 2 activitățile specifice obiectivului 1 corespunzător etapei 2015.

Capitolul 3 a cuprins o introducere referitoare la mai multe aspecte generale despre: vehicule autonome, motivația dezvoltării unui astfel de proiect, adaptive cruise control, plutoanele de autovehicule și sistemele de comunicații între vehicule.

În capitolul 4 au fost determinate matematic mai multe modele pentru vehicule din diverse puncte de vedere: modele care surprind dinamica laterală a vehiculului (care vor fi utilizat pentru schimbarea direcției de mers sau a benzii de circulație), modele care surprind dinamica longitudinală a vehiculului (care vor fi utilizate pentru controlul vitezei longitudinale - cruise control), modele pentru roboții mobili (ce vor fi utilizate pentru experimentele de timp real) și două modele ale unor plutoane de autovehicule.

Capitolul 5 a făcut referire la modelarea rețelelor de comunicații și a problemelor introduse de acestea utilizând două tipuri de modele pentru întârzierile variabile în timp: modelare utilizând incluziuni politopice și modelare considerându-le ca perturbații.

În capitolul 6 s-a discutat despre posibilitatea implementării reguletoarelor predictive în timp real, tipuri de arhitectura hardware folosite (implementarea reguletoarelor predictive cu FPGA, implementarea reguletoarelor predictive cu microcontrollere), particularități

constructive ale roboților mobili utilizați în sisteme multi-agent sau de tip pluton și despre modelarea restricțiilor hardware ca partiții poliedrale ale spațiului stărilor

Capitolul 7 a cuprins trei metode de control pentru a se obține un comportament cât mai eficient al modelelor vehiculelor utilizate în plutoanele de autovehicule: metoda de control ce utilizează un regulator PID, regulatoare bazate pe metoda sliding mode control (SMC), regulatoare bazate pe control predictiv (MPC). În ultimul subcapitol au fost prezentate câteva modele de vehicule utilizate în controlul predictiv distribuit.

Prin îmbinarea modelelor prezentate anterior se obține un model care surprinde atât dinamica laterală și dinamica longitudinală a autovehiculelor, cât și restricțiile introduse de comunicațiile dintre autovehiculele dintr-un pluton, precum și restricțiile hardware introduse de implementarea regulatoarelor predictive distribuite în aplicații de timp real.

Cunoscând toate informațiile prezentate pe parcursul tuturor capitolele din acest raport putem concluziona că a fost îndeplinit obiectivul acestei etape de desfășurare a proiectului.

Etapa II

Dezvoltarea strategiilor de control predictiv bazate pe agenți

II.1 Obiectivele etapei

Obiectiv 2: Dezvoltarea strategiilor de control predictiv bazate pe agenți

- **Activitate 2.1.** Dezvoltarea de strategii predictive de control folosind agenți bazate pe PLF-uri utilizând modelul distribuit realist dezvoltat în Etapa 1 și considerând perturbațiile din trafic
- **Activitate 2.2.** Dezvoltarea de emulatoare în Matlab pentru plutoanele de vehicule cooperante
- **Activitate 2.3.** Verificarea stabilității plutonului de vehicule cooperante bazate pe multiagenți
- **Activitate 2.4.** Analiza strategiei de control predictive bazate pe agenți
- **Livrabil 3:** Raport cu descrierea emulatorului Matlab pentru plutoanele de vehicule.
- **Livrabil 4:** Raport cu descrierea strategiei de control predictive bazată pe agenți.
- **Livrabil 5:** Analiza stabilității plutonului de vehicule cooperante.

II.2 (A2.1) Dezvoltarea de strategii predictive de control folosind agenți bazate pe PLF-uri utilizând modelul distribuit realist dezvoltat în Etapa 1 și considerând perturbațiile din trafic

II.2.1 Dezvoltarea unui algoritm de control predictiv distribuit non-cooperativ pentru urmărirea referinței

În această secțiune este prezentat un algorithm de control predictiv distribuit DMPC (eng. Distributed Model Predictive Control) pentru urmărirea unei referințe treaptă. În acest scop, este utilizat un model augmentat (i.e., un integrator este integrat în bucla de control) și vectorul de stare augmentat conține incrementele stărilor și eroarea între referință și ieșirea predictată. Algoritmul dezvoltat este testat într-un experiment în timp real pe un proces cu patru rezervoare și performanța este comparată cu varianta centralizată a algoritmului.

II.2.1.1 Introducere

Controlul predictiv distribuit este o alternativă de control utilizată în industrie, atunci când se dorește evitarea poverii de calcul a controlului centralizat [113]. Conform studiului din [114], metodele DMPC pot fi împărțite în: *i*) DMPC *cooperativ*, în care fiecare regulator local minimizează o funcție de cost globală sau *ii*) DMPC *non-cooperativ*, dacă o funcție de cost locală este minimizată. De asemenea, în [115] algoritmii DMPC sunt clasificați depinzând de topologia rețelei de comunicare în algoritmi: *total conectați* sau *parțial conectați*. Din punct de vedere a cantității de informație comunicată între regulatoarele locale, algoritmii pot fi: *i*) *non-iterativi*, dacă informația este schimbată o singură dată într-o perioadă de eșantionare sau *ii*) *iterativi*, dacă regulatoarele locale fac schimb de informații de mai multe ori într-o perioadă de eșantionare.

Urmărirea unei referințe fără eroare staționară este una dintre cele mai comune cerințe de control. În [116–118] este prezentată o metodă centralizată pentru urmărirea referinței. Ideea este să se folosească un integrator ca și model pentru perturbații, pentru a se ține cont de erorile de modelare. Mai mult, valorile de regim staționar ale stărilor și perturbațiilor impuse sunt folosite în scopul urmăririi referinței fără eroare staționară. Această metodă, cu toate că este eficientă, are nevoie de un model centralizat pentru a calcula valorile de regim staționar care nu este disponibil într-un context distribuit.

În [119], sunt prezentate câteva soluții pentru problema urmăririi referinței fără eroare staționară, toate având la bază un integrator în bucla de control. Aceste soluții se bazează pe: *i*) introducerea unei perturbații artificiale care va fi estimată împreună cu stările sistemului [120] *ii*) folosirea unui integrator ca un model intern pentru referință care va fi introdus în mod direct în bucla de control [121] sau *iii*) folosirea unui model *de viteză* în care vectorul de stare este augmentat cu incrementele stării și eroarea [122].

Metoda propusă în această secțiune este o continuare a studiului prezentat în [123], care introduce un algoritm DMPC non-cooperativ pentru un sistem cu doi agenți (i.e., care

poate fi descompus în două sub-sisteme). Problema de reglare (i.e., urmărirea unei referințe constante) a fost rezolvată cu ajutorul unui model centralizat. Acesta soluție constă în folosirea unui model centralizat al sistemului pentru a obține valorile de regim staționar pentru stările și intrările sistemului prin impunerea valorilor de regim staționar pentru ieșiri. Metoda propusă, deși eficientă în simulare, pornește de la premiza că modelul este identic cu sistemul și nu rămâne validă în experimentele din realitate.

Astfel, în continuare, este propusă o altă metodă DMPC non-cooperativă potrivită pentru urmărirea unei referințe constante. Ideea a pornit de la formularea centralizată dată în [122], în care un model ‘*de viteză*’ este folosit, introducând un integrator artificial în sistem. În acest fel, erorile de modelare nu vor mai influența rezultatele în buclă închisă, obținându-se eroare staționară nulă în experimentele reale.

II.2.1.2 Modelul *de viteză* folosit în proiectarea regulatorului predictiv

Fie un sistem liniar invariant în timp, alcătuit din două sub-sisteme cuplate dinamic prin intrări și sub influența restricțiilor pe variabilele de intrare și stare. Fiecare sub-sistem este descris de următoarea dinamică:

$$\begin{aligned} x_{pi}(k+1) &= A_{pi}x_{pi}(k) + B_{pii}u_i(k) + B_{pij}u_{ni}(k) \\ y_i(k) &= C_{pi}x_{pi}(k) \end{aligned} \quad (\text{II.1})$$

unde k este eșantionul de timp, $u_{pi} \in R^{m_i}$ și $y_i \in R^{q_i}$ sunt variabilele de intrare/ieșire ale sub-sistemelor și $x_{pi} \in R^{n_i}$ sunt stările fiecărui sub-sistem, cu $i = \overline{1, 2}$, $j = \overline{1, 2}$, $i \neq j$. Indicele inferior ni notează agentul vecin pentru agentul i (i.e., $u_{n1} = u_2$ și $u_{n2} = u_1$). Se impun următoarele restricții pentru variabilele de intrare și stare:

$$x_{pi} \in X_i, u_i \in U_i, i = \overline{1, 2} \quad (\text{II.2})$$

unde X_i și U_i cu $i = \overline{1, 2}$ sunt definite prin seturi de inecuații liniare.

Folosind metodologia descrisă în [122], operația de diferență este aplicată în ambele membre ale modelului (II.50), obținând:

$$\begin{aligned} \underbrace{\begin{bmatrix} \Delta x_{pi}(k+1) \\ y_i(k+1) \end{bmatrix}}_{x_i(k+1)} &= \underbrace{\begin{bmatrix} A_{pi} & O_{q_i \times n_i}^T \\ C_{pi} & A_{pi} \end{bmatrix}}_{A_i} \underbrace{\begin{bmatrix} \Delta x_{pi}(k) \\ y_i(k) \end{bmatrix}}_{x_i(k)} \\ &+ \underbrace{\begin{bmatrix} B_{pii} \\ C_{pi} B_{pii} \end{bmatrix}}_{B_{ii}} \Delta u_i(k) + \underbrace{\begin{bmatrix} B_{pij} \\ C_{pi} B_{pij} \end{bmatrix}}_{B_{ij}} \Delta u_{ni}(k) \\ y_i(k) &= \underbrace{\begin{bmatrix} O_{q_i \times n_i} & I_{q_i} \end{bmatrix}}_{C_i} \begin{bmatrix} \Delta x_{pi}(k) \\ y_i(k) \end{bmatrix} \end{aligned} \quad (\text{II.3})$$

ce va fi folosit în proiectarea regulatorului DMPC. Este important de remarcat că noile intrări în modelul în spațiul stărilor *velocity-form* sunt $\Delta u_i(k)$ and $\Delta u_{ni}(k)$. Indicele superior T definește operația de transpunere matricială, I este matricea identitate și O este matricea nulă. Matricile cu dimensiuni adecvate A_i , B_{ii} , B_{ij} și C_i cu $i = \overline{2, N}$, $j = i - 1$ descriu modelul discret al sub-sistemelor în formulare *de viteză*.

Modelul augumentat (II.23) poate fi scris într-o formă compactă ca:

$$\begin{cases} x_i(k+1) = A_i x_i(k) + B_{ii} \Delta u_i(k) + B_{ij} \Delta u_{ni}(k) \\ y_i(k) = C_i x_i(k) \end{cases} \quad (\text{II.4})$$

cu $i = \overline{1, 2}$, $j = \overline{1, 2}$, $i \neq j$.

II.2.1.3 Problema de optimizare non-cooperativă DMPC

Pentru sistemul liniar, invariant în timp, compus din doua sub-sisteme definite prin (II.50), se proiectează o arhitectură de control non-cooperativ distribuit în care cei doi agenți corespunzători celor două sub-sisteme comunică între ei. Astfel, fiecare sub-sistem i este controlat de un agent local (i.e., un regulator predictiv) care, la fiecare perioadă de eșantionare k are disponibilă doar informația corespunzătoare sub-sistemului propriu.

Pornind de la modelul *de viteză* al sub-sistemului i (II.4) și starea corespunzătoare x_i , este calculată variabila de control Δu_i . La fiecare perioadă de eșantionare, fiecare agent rezolvă o problemă de optimizare de dimensiuni reduse, impunând o traiectorie fixată pentru vecinul său. Obiectivul de control al fiecărui agent este ca fiecare sub-sistem să urmărească o referință constantă cu satisfacerea restricțiilor impuse.

Folosind modelul *de viteză* (II.4) pentru proiectarea reguletoarelor predictive, se încorporează în mod natural o acțiune integrală în algoritmul de control.

Ideia principală este că fiecare sub-sistem rezolvă o problemă de optimizare considerând că dinamica este dată de (II.4) și secvența de control viitoare corespunzătoare vecinului $\Delta u_{ni}(k+l)$ sub-sistemului i este cunoscută pe întreg orizontul de comandă N_c , $l = \overline{0, N_c - 1}$. Astfel, singura informație necesară agentului i la timpul k este primită de la agentul vecin ni , $\Delta u_{ni}(k+l)$. Pentru a se obține un algoritm eficient din punct de vedere computațional, secvența de intrare viitoare corespunzătoare vecinului se calculează ca în [123] folosind ultimele $N_c - 1$ elemente din secvența de control optimă calculată la eșantionul precedent $k - 1$. Pentru a se păstra aceeași lungime a vectorilor, se dublează ultimul element din vector. Acesta este similar cu un orizont de control mai scurt cu o unitate pentru vecin (i.e., $N_c - 1$ în loc de N_c).

$$\Delta U_{ni}(k) = \begin{bmatrix} \Delta u_{ni}^*(k|k-1) \\ \vdots \\ \Delta u_{ni}^*(k+N_c-2|k-1) \\ \Delta u_{ni}^*(k+N_c-2|k-1) \end{bmatrix} \quad (\text{II.5})$$

Astfel, la fiecare moment $k-1$ fiecare agent i , $i = \overline{1, 2}$ transmite celuilalt agent secvența optimă $\Delta U_i^*(k-1)$ care va fi folosită la momentul k pentru a crea $\Delta U_{ni}(k)$ (i.e., ecuația (II.5)).

Pentru fiecare agent i , o funcție de cost locală este definită ca:

$$\begin{aligned} J_i(x_i(k), \Delta U_i(k), \Delta U_{ni}(k)) = \\ (R_{si} - Y_i)^T (R_{si} - Y_i) + \Delta U_i(k)^T R_i \Delta U_i(k) \end{aligned} \quad (\text{II.6})$$

bazată pe ieșirea predictată:

$$Y_i = [y_i(k+1|k) \ \dots \ y_i(k+N_p|k)]^T$$

și secvența de intrare proprie fiecărui sub-sistem

$$\Delta U_i(k) = \underbrace{[\Delta u_i(k|k) \ \dots \ \Delta u_i(k + N_c - 1|k)]^T}_{N_p}$$

cu $i = \overline{1, N}$, orizontul de predicție N_p și orizontul de control N_c ($N_c \leq N_p$). Matricile de ponderare a intrării sunt definite ca $R_i = \alpha I_{N_c}$, $i = \overline{1, 2}$, $\alpha \geq 0$.

Folosind modelul *de viteză* (II.4), se calculează predicția variabilelor de stare și ieșire:

$$\begin{aligned} x_i(k + l|k) = & A_i^l x_i(k) + A_i^{l-1} B_{ii} \Delta u_i(k|k) + \dots + \\ & A_i^{l-l} B_{ii} \Delta u_i(k + l - 1|k) + A_i^{l-1} B_{ij} \Delta u_{ni}(k|k) + \\ & + \dots + A_i^{l-l} B_{ij} \Delta u_{ni}(k + l - 1|k) \\ & l = \overline{1, N_p} \end{aligned} \quad (\text{II.7})$$

$$\begin{aligned} y_i(k + l|k) = & C_i A_i^l x_i(k) + C_i A_i^{l-1} B_{ii} \Delta u_i(k|k) \\ & + \dots + C_i A_i^{l-l} B_{ii} \Delta u_i(k + l - 1|k) + \\ & C_i A_i^{l-1} B_{ij} \Delta u_{ni}(k|k) + \dots + \\ & C_i A_i^{l-l} B_{ij} \Delta u_{ni}(k + l - 1|k) \\ & l = \overline{1, N_p} \end{aligned} \quad (\text{II.8})$$

O remarcă importantă este faptul că funcția cost (II.25) depinde de starea măsurată la momentul curent $x_i(k)$ și de secvența de intrare viitoare a vecinului $\Delta U_{ni}(k)$ pentru că predictorul ieșirii Y_i , $i = \overline{1, N}$ este calculat în formă matricială pornind de la (II.28):

$$\begin{aligned} Y_i = & \tilde{A}_i x_i(k) + \tilde{B}_{ii} \Delta U_i(k) + \tilde{B}_{ij} \Delta U_{ni}(k) \\ & i = \overline{1, 2}, \quad j = \overline{1, 2}, \quad i \neq j \end{aligned} \quad (\text{II.9})$$

unde

$$\begin{aligned} \tilde{A}_i = & [C_i A_i \quad C_i A_i^2 \quad \dots \quad C_i A_i^{N_p}]^T \\ \tilde{B}_{ii} = & \begin{bmatrix} C_i B_{ii} & 0 & \dots & 0 \\ C_i A_i B_{ii} & C_i B_{ii} & \dots & 0 \\ \vdots & \vdots & \vdots & \vdots \\ C_i A_i^{N_p-1} B_{ii} & C_i A_i^{N_p-2} B_{ii} & \dots & C_i A_i^{N_p-N_c} B_{ii} \end{bmatrix} \end{aligned}$$

și \tilde{B}_{ij} este obținut similar cu \tilde{B}_{ii} .

După substituția predictorului (II.29) în funcția cost (II.25) și calculele matriciale corespunzătoare, se obține următoarea formă explicită a funcției cost:

$$\begin{aligned} J_i(x_i(k), \Delta U_i, \Delta U_{ni}) = & (R_{si} - \tilde{A}_i x_i(k))^T (R_{si} - \tilde{A}_i x_i(k)) + 2 \Delta U_i^T \tilde{B}_{ii}^T \tilde{B}_{ij} \Delta U_{ni} \\ & - 2 \Delta U_i^T \tilde{B}_{ii}^T [R_{si} - \tilde{A}_i x_i(k)] - 2 \Delta U_{ni}^T \tilde{B}_{ij}^T [R_{si} - \tilde{A}_i x_i(k)] \\ & + 2 \Delta U_i^T (\tilde{B}_{ii}^T \tilde{B}_{ii} + R_i) \Delta U_i + \Delta U_{ni}^T (\tilde{B}_{ij}^T \tilde{B}_{ij}) \Delta U_{ni} \end{aligned} \quad (\text{II.10})$$

cu $i = \overline{1,2}$, $j = \overline{1,2}$, $i \neq j$.

După cum sa menționat mai devreme, în (II.30) variabila liberă este $\Delta U_i(k)$, $i = \overline{1,2}$. Dacă nu sunt impuse restricții pentru variabila de control sau ieșiri, soluția optimală se obține egalând derivata de ordin unu a (II.30) cu zero, obținându-se:

$$\Delta U_i^*(k) = (\tilde{B}_{ii}^T \tilde{B}_{ii} + R_i)^{-1} \tilde{B}_{ii}^T [R_{si} - \tilde{A}_i x_i(k)] - (\tilde{B}_{ii}^T \tilde{B}_{ii} + R_i)^{-1} \tilde{B}_{ii}^T \tilde{B}_{ij} \Delta U_{ni}, \quad (\text{II.11})$$

cu $i = \overline{1,2}$, $j = \overline{1,2}$, $i \neq j$. Pe de altă parte, una din avantajele controlului predictiv este impunerea de restricții direct în funcția de optimizare. Restricțiile sunt impuse pentru variația incrementală ale variabilelor :

$$\Delta U_i^{min} \leq \Delta U_i(k) \leq \Delta U_i^{max} \quad (\text{II.12})$$

on the amplitude of the control variables:

$$U_i^{min} \leq U_i(k) \leq U_i^{max} \quad (\text{II.13})$$

and on the outputs:

$$Y_i^{min} \leq Y_i(k) \leq Y_i^{max} \quad (\text{II.14})$$

cu $i = \overline{1,2}$, unde $U_i^{min} = [u_i^{min} \dots u_i^{min}]^T$ este un vector coloană cu N_c elemente care conține limitele minime impuse pentru fiecare element din fereastra de predicție. O definiție similară este folosită pentru U_i^{max} , ΔU_i^{min} , ΔU_i^{max} , Y_i^{min} și Y_i^{max} .

Potrivit cu formularea pătratică, (II.13) este despărțită în doua inegalități:

$$\begin{aligned} -U_i(k) &\leq -U_i^{min} \\ U_i(k) &\leq U_i^{max} \end{aligned} \quad (\text{II.15})$$

cu $i = \overline{1,2}$. Aceași notație patrată este folosită și pentru (II.32) și (II.14).

Următorul pas este parametrizarea restricției pentru variabila de control, în variabila ΔU_i , $i = \overline{1,2}$ obținând:

$$\begin{aligned} &\underbrace{\begin{bmatrix} u_i(k|k) \\ u_i(k+1|k) \\ \vdots \\ u_i(k+N_c-1|k) \end{bmatrix}}_{U_i(k)} = \underbrace{\begin{bmatrix} 1 \\ 1 \\ \vdots \\ 1 \end{bmatrix}}_{M_{ij}} u_i(k-1) \\ &+ \underbrace{\begin{bmatrix} 1 & 0 & 0 & \dots & 0 \\ 1 & 1 & 0 & \dots & 0 \\ \vdots & & & & \\ 1 & 1 & 1 & \dots & 1 \end{bmatrix}}_{M_{ii}} \underbrace{\begin{bmatrix} \Delta u_i(k|k) \\ \Delta u_i(k+1|k) \\ \vdots \\ \Delta u_i(k+N_c-1|k) \end{bmatrix}}_{\Delta U_i} \end{aligned} \quad (\text{II.16})$$

$i = \overline{1,2}$, $j = \overline{1,2}$, $i \neq j$.

Substituind (II.16) în (II.15) se obține următoarea formă compactă:

$$\begin{aligned} -(M_{ij}u_i(k-1) + M_{ii}\Delta U_i(k)) &\leq -U_i^{min} \\ (M_{ij}u_i(k-1) + M_{ii}\Delta U_i(k)) &\leq U_i^{max} \end{aligned} \quad (II.17)$$

Restricțiile pentru variabila de ieșire sunt parametrizate folosind definiția din (II.29) obținând:

$$\begin{aligned} -(\tilde{A}_i x_i(k) + \tilde{B}_{ii}\Delta U_i(k)) &\leq -Y_i^{min} + \tilde{B}_{ij}\Delta U_{ni}(k) \\ (\tilde{A}_i x_i(k) + \tilde{B}_{ii}\Delta U_i(k)) &\leq Y_i^{max} - \tilde{B}_{ij}\Delta U_{ni}(k) \end{aligned} \quad (II.18)$$

cu $i = \overline{1,2}$, $j = \overline{1,2}$, $i \neq j$.

Toate restricțiile (II.32), (II.17), (II.18) pot fi rescrise în formă matricială ca:

$$\begin{bmatrix} -I \\ I \\ -M_{ii} \\ M_{ii} \\ -\tilde{B}_{ii} \\ \tilde{B}_{ii} \end{bmatrix} \Delta U_i = \begin{bmatrix} -\Delta U_i^{min} \\ \Delta U_i^{max} \\ -U_i^{min} + M_{ij}u_i(k-1) \\ U_i^{max} - M_{ij}u_i(k-1) \\ -Y_i^{min} + \tilde{A}_i x_i(k) + \tilde{B}_{ij}\Delta U_{ni} \\ Y_i^{max} - \tilde{A}_i x_i(k) - \tilde{B}_{ij}\Delta U_{ni} \end{bmatrix} \quad (II.19)$$

cu $i = \overline{1,2}$, $j = \overline{1,2}$, $i \neq j$ și I matricea identitate.

Astfel, problema de optimizare cu restricții este formulată astfel: să se găsească minimumul funcției de cost (II.30) cu restricțiile definite în (II.19).

II.2.1.4 Pluton de vehicule descrise prin model dublu integrator

Controlul distribuit aplicat sistemelor autonome multi-agent a provocat un interes crescut în comunitatea de control datorită avantajelor pe care le prezintă, printre care robustețe și flexibilitate. În [124], este prezentată o vedere de ansamblu asupra contribuțiilor recente din acest domeniu, pe următoarele direcții: *i*) consens, i.e., conducerea unui grup de agenți către un scop comun, *ii*) controlul distribuit al formației de vehicule, i.e., obținerea unei configurații geometrice dorite cu ajutorul schimbului de informații între agenți, în prezența sau fără un lider al grupului și *iii*) controlul distribuit bazat pe estimare locală, în care o informație globală necunoscută este estimată local de fiecare agent. În cazul consensului și a controlului distribuit a formației de vehicule, obiectivul este urmărirea referinței impuse cu ajutorul comunicării între agenți.

Stabilitatea formației de vehicule în absența unui lider, poate fi realizată prin diferite metode ca: *i*) teoria matricială, unde valorile proprii ale matricii sistemului în buclă închisă sunt investigate, *ii*) metoda bazată pe funcții Lyapunov în care stabilitatea este definită utilizând o funcție Lyapunov sau *iii*) control predictiv (MPC, eng. model predictive control) sau controlul bazat pe principiul orizontului alunecător (RHC, eng. receding horizon control), care tratează în esență o problemă de optimizare cu orizont finit.

În [125] este prezentată o strategie de coordonare distribuită bazată pe teoria grafurilor, care asigură că un grup de agenți mobili rămân conectați obținând controlul formației.

Controlul unei formații lider-urmăritor bazat pe cooperarea între vehicule dotate cu vedere artificială, în care problema de optimizare are restricții de comunicare și vedere este propus în [126].

În [127] este prezentată o problemă de control distribuit pentru coordonarea vehiculelor, folosind o formulare bazată pe funcții Lyapunov iar în [128] este propusă o strategie de control predictiv pentru coordonarea vehiculelor. Ideea este folosirea problemei de optimizare din formularea MPC, pentru a asigura stabilitatea formației, prin minimizarea de către fiecare agent a unei funcții de cost globale. În [105] este descris un algoritm de control distribuit lider-urmăritor folosind concepte din teoria grafurilor. Metoda folosește ideile generale din domeniul coordonării sistemelor multi-agent, pentru a permite agenților cu dinamici diferite urmărirea liderului.

Depinzând de definiția folosită pentru cuplajele dintre sub-sisteme, schemele de control predictiv distribuit (DMPC, eng. distributed model predictive control) se clasifică în strategii în care interacția dintre vehicule este introdusă prin intermediul: *i*) un termen specific în funcția cost [102] sau *ii*) o restricție specifică inpusă pentru stările sistemului [103].

Între toate obiectivele de control formulate în sistemele multi-agent, cea mai comună este controlul unui pluton de vehicule care se deplasează în formație liniară. În [104] este introdusă o strategie de control predictiv robust, aplicat pentru un sistem distribuit cu structură în lanț. Acest tip de aplicație este specific unui pluton de vehicule în formație în linie, care trebuie să mențină o distanță constantă între vehicule. Pentru a se asigura satisfacerea obiectivului, o restricție de robustețe este introdusă în funcție de optimizare și capabilități de comunicare fără fir (eng. wireless) sunt introduse.

În această secțiune este prezentată o strategie de control DMPC pentru un pluton de vehicule în formație în linie. Algoritmul utilizat a fost dezvoltat inițial [30, 123] pentru un sistem cu doi agenți, cuplați prin intrări. Această metodologie a fost reformulată pentru o aplicație cu agenți multipli într-o arhitectură în lanț (i.e., cuplarea și comunicarea între agenți este unidirecțională). O aplicație potrivită pentru o astfel de arhitectură este controlul unei formații de vehicule lider-urmăritor (eng. leader-follower formation control), în care vehiculul lider comunică informațiile relevante către primul vehicul urmăritor, care la rândul lui va acționa ca un ‘lider relativ’ pentru al doilea vehicul urmăritor s.a.m.d. Rezultatele s-au obținut într-un scenariu în care comunicația între vehicule este fără întârzieri (i.e., informația comunicată de la vehiculul predecesor, necesară în strategia de control este recepționată în aceeași perioadă de eșantionare). De asemenea s-au comparat două soluții pentru problema de optimizare (cu și fără restricții).

Modelarea vehiculelor și formularea problemei de optimizare Fie o rețea de N vehicule modelate cu o dinamică de dublu integrator (i.e., o masă într-un spațiu uni-dimensional, care influențată de o forță de intrare variantă în timp), folosite în mod comun în literatură [104, 105, 129–133].

Fiecare sub-sistem urmăritor $i \in \{2 \dots N\}$ este descris de următorul model în timp

discret [104]:

$$\begin{aligned} x_i(k+1) &= \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix} x_i(k) + \\ &\begin{bmatrix} -T_s^2/2 \\ -T_s \end{bmatrix} u_i(k) + \begin{bmatrix} T_s^2/2 \\ T_s \end{bmatrix} u_{ni}(k) \\ y_i(k) &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x_i(k) \end{aligned} \quad (\text{II.20})$$

unde T_s este perioada de eșantionare și indicele inferior $ni \in \{1 \dots N - 1\}$ este folosit pentru indicarea vehicului vecin (i.e., $ni = 1$ este liderul și vecinul celui de-al doilea vehicul din lanț, în timp ce $ni = N - 1$ este predecesorul și liderul pentru ultimul vehicul din formație). Fiecare vehicul i poate măsura distanța d_i și viteza relativă \dot{d}_i raportată la vehiculul predecesor și obiectivul de control pentru toate vehiculele urmăritor din formația în linie este urmărirea ‘liderului relativ’ cu o spațiere constantă. Stările din modelul (II.20) sunt eroarea față de distanța dintre vehicule S , definită ca $e_i = d_i - S$ și \dot{e}_i , $i = \overline{2, N}$. Mărimea de comandă u_i este accelerația vehiculelor. Pentru fiecare vehicul urmăritor, minimizează ambele stări în zero, asigurând că fiecare vehicul se deplasează cu aceeași viteză ca și precedentul și că păstrează distanța impusă S .

Vehiculul lider al formației este controlat cu agentul $i = 1$. Obiectivul de control al liderului este să urmărească o traiectorie de referință impusă pentru poziție. Vehiculul lider poate accesa doar stările proprii, nu primește informații despre stările locale ale vehiculelor urmăritor și comunică informațiile (i.e., traiectoria de comandă proprie) doar către vehiculul urmăritor imediat (i.e., agentul $i = 2$). Astfel, modelul dublu integrator în timp discret al vehiculului lider este:

$$\begin{aligned} x_i(k+1) &= \begin{bmatrix} 1 & T_s \\ 0 & 1 \end{bmatrix} x_i(k) + \begin{bmatrix} T_s^2/2 \\ T_s \end{bmatrix} u_i(k) \\ y_i(k) &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x_i(k) \end{aligned} \quad (\text{II.21})$$

unde stările sunt poziția p_i și viteza v_i , $i = 1$ iar u_i este accelerația.

Fiecare vehicul urmăritor i , $i = \overline{2, N}$ are acces doar la informația locală, primește traiectoria de comandă calculată de vehiculul predecesor (i.e., $i - 1$) folosită pentru a rezolva problema de optimizare într-o manieră distribuită și trimite secvența de comandă calculată către vehiculul succesiv (i.e., $i + 1$). Într-un caz special, ultimul vehicul din formație (i.e., $i = N$) doar primește comanda de la predecesor și nu comunică cu celelalte vehicule.

În figura II.5 este prezentată o diagramă schematică a unui pluton de N vehicule în formație în linie. Distanța între vehiculele V_i , $i = \overline{1, N}$ este notată cu d_i , $i = \overline{1, N - 1}$ și semnalele comunicate sunt marcate cu săgeți punctate (i.e., secvența de comandă calculată local u_i , $i = \overline{1, N - 1}$).

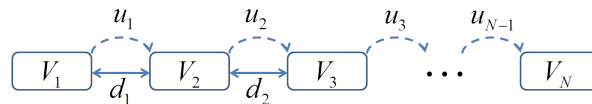


Figura II.1: Diagramă schematică a unui pluton de N vehicule în formație în linie.

Descrierea algoritmului DMPC În această secțiune este prezentat pe scurt algoritmul propus în [30, 123]. Fie un sistem liniar, invariant în timp, compus din câteva sub-sisteme cuplate dinamic prin intrări. Fiecare sub-sistem este descris de următoarea dinamică:

$$\begin{aligned} x_{pi}(k+1) &= A_{pi}x_{pi}(k) + B_{pii}u_i(k) + B_{pij}u_{ni}(k) \\ y_i(k) &= C_{pi}x_{pi}(k) \end{aligned} \quad (\text{II.22})$$

unde k este indicele de timp discret, $u_{pi} \in R$ și $y_i \in R^2$ sunt variabilele de intrare, respectiv ieșire corespunzătoare fiecărui sub-sistem și $x_{pi} \in R^2$ sunt stările sub-sistemelor, cu $i = \overline{2, N}$.

Pentru problema de control a unui pluton de vehicule propusă în acest raport, obiectivele de control sunt definite în funcție de tipul vehiculului, după cum urmează:

- pentru vehiculul lider al plutonului, se impune urmărirea unei referințe de poziție fără eroare staționară (i.e., definită ca un semnal rampă) și
- pentru vehiculele următor, stările sub-sistemului $e_i = d_i + S$ și \dot{e}_i , $i = 2, N$ trebuie minimizate în zero.

Cu toate acestea, ambele obiective de control sunt interconectate deoarece sub-sistemele sunt cuplate prin intrări (vezi figura II.5) și este necesar un schimb de informații. Faptul că liderul își îndeplinește obiectivul de control, asigură satisfacerea obiectivelor vehiculelor următor. Acest rezultat vine din faptul că, intrarea optimală calculată de lider este comunicată primului vehicul următor, care folosește această informație în problema de optimizare locală. Apoi, primul vehicul următor comanda optimală proprie calculată local către cel de-al doilea vehicul următor (i.e., al treilea vehicul din lanț), procedura repetându-se pentru fiecare vehicul din pluton.

Considerând cazul particular al unui sistem dublu integrator și folosindu-ne de teorema valorii finale [134], un integrator în regulator este suficient pentru a satisface obiectivele de control. În acest raport, un integrator este introdus în structura de control prin reformularea în model de viteză.

Folosind metodologia descrisă în [122], modelul augmentat cu noile variabile de stare $x_i(k) = [\Delta x_{pi}(k)^T y_i(k)]^T$ este obținut aplicând operația de diferență în ambele membre ale modelului (II.50):

$$\begin{aligned} \underbrace{\begin{bmatrix} \Delta x_{pi}(k+1) \\ y_i(k+1) \end{bmatrix}}_{x_i(k+1)} &= \underbrace{\begin{bmatrix} A_{pi} & O_{q_i \times n_i}^T \\ C_{pi}A_{pi} & 1 \end{bmatrix}}_{A_i} \underbrace{\begin{bmatrix} \Delta x_{pi}(k) \\ y_i(k) \end{bmatrix}}_{x_i(k)} \\ &+ \underbrace{\begin{bmatrix} B_{pii} \\ C_{pi}B_{pii} \end{bmatrix}}_{B_{ii}} \Delta u_i(k) + \underbrace{\begin{bmatrix} B_{pij} \\ C_{pi}B_{pij} \end{bmatrix}}_{B_{ij}} \Delta u_{ni}(k) \\ y_i(k) &= \underbrace{\begin{bmatrix} O_{q_i \times n_i} & I_{q_i} \end{bmatrix}}_{C_i} \begin{bmatrix} \Delta x_{pi}(k) \\ y_i(k) \end{bmatrix} \end{aligned} \quad (\text{II.23})$$

ce va fi folosit în proiectarea regulatorului DMPC. Este important de remarcat că noile intrări în modelul în spațiul stărilor *velocity-form* sunt $\Delta u_i(k)$ and $\Delta u_{ni}(k)$. Indicele superior T definește operația de transpunere matricială, I este matricea identitate și O este

matricea nulă. Matricile cu dimensiuni adecvate A_i , B_{ii} , B_{ij} și C_i cu $i = \overline{2, N}$, $j = i - 1$ descriu modelul discret al sub-sistemelor în formulare *de viteză*.

O altă remarcă este faptul că vehiculul lider este un caz special de sub-sistem fără vecini. Astfel, modelul liderului este definit similar cu (II.23), pornind de la (II.38), în care lipsește termenul depinzând de intrarea vecinului, obținându-se:

$$\begin{aligned} x_i(k+1) &= A_i x_i(k) + B_{ii} u_i(k) \\ y_i(k) &= C_i x_i(k) \end{aligned} \quad (\text{II.24})$$

unde indicele inferior $i = 1$ notează liderul plutonului.

O arhitectură de control distribuit non-cooperativă este definită pentru un sistem liniar invariant în timp, compus din multiple sub-sisteme descrise prin ecuațiile (II.23), (II.24). Toți agenții implicați au o comunicație uni-direcțională între ei (i.e., agentul i primește informații doar de la agentul $i - 1$ cu $i = \overline{2, \dots, N}$). Fiecare sub-sistem i este controlat de un agent local (un regulator predictiv), în care informația locală proprie fiecărui sub-sistem și informația primită de la vecin este disponibilă la fiecare eșantion de timp k . O acțiune integrală este introdusă în sistemul de control prin folosirea modelului *de viteză* al sub-sistemului (II.23), (II.24) în proiectarea regulatorului predictiv.

Pentru fiecare agent i , o funcție de cost locală este definită ca:

$$\begin{aligned} J_i(x_i(k), \Delta U_i(k), \Delta U_{ni}(k)) = \\ (R_{si} - Y_i)^T (R_{si} - Y_i) + \Delta U_i(k)^T R_i \Delta U_i(k) \end{aligned} \quad (\text{II.25})$$

bazată pe ieșirea predictată:

$$Y_i = [y_i(k+1|k) \ \dots \ y_i(k+N_p|k)]^T$$

și secvența de intrare proprie fiecărui sub-sistem

$$\Delta U_i(k) = \underbrace{[\Delta u_i(k|k) \ \dots \ \Delta u_i(k+N_c-1|k)]^T}_{N_p}$$

cu $i = \overline{1, N}$, orizontul de predicție N_p și orizontul de control N_c ($N_c \leq N_p$). Matricile de ponderare a intrării sunt definite ca $R_i = \alpha I_{N_c}$, $i = \overline{1, 2}$, $\alpha \geq 0$.

Pornind de la valoarea semnalului de referință $r_i(k)$ la momentul de timp k , scopul este minimizarea erorii dintre ieșirea viitoare și semnalul de referință, în fereastra orizontului de predicție (i.e., primul termen din (II.25)), în timp ce al doilea termen ponderează valoarea $\Delta U_i(k)$ când (II.25) este minimizată. Astfel, în funcție de tipul vehiculului, sunt impuși următorii vectori de referință $R_{si} \in R^{N_p}$, $i = \overline{1, N}$:

- pentru vehiculul lider, vectorul care conține informația impusă ca referință în fereastra orizontului de predicție $R_{s1} \in R^{N_p}$, $i = 1$ este proiectat ca fiind un semnal rampă cu aceeași pantă m ca valoarea semnalului de referință $r_i(k)$ la momentul de timp k :

$$R_{s1}(k+l|k) = l * m + r_1(k) \quad l = \overline{1, N_p} \quad (\text{II.26})$$

- pentru vehiculele următor, semnalul de referință $r_i(k)$ este 0 pe întregul experiment (întrucât scopul este minimizarea stărilor în zero) și vectorul care conține informația

impusă ca referință în fereastra orizontului de predicție $R_{si} \in R^{N_p}$, $i = \overline{2, N}$ devine:

$$R_{si} = \underbrace{[1 \ \dots \ 1]^T}_{N_p} r_i(k) = \underbrace{[0 \ \dots \ 0]^T}_{N_p}$$

Folosind modelul *de viteză* (II.23), se calculează predicția variabilelor de stare și ieșire:

$$\begin{aligned} x_i(k+l|k) = & A_i^l x_i(k) + A_i^{l-1} B_{ii} \Delta u_i(k|k) + \dots + \\ & A_i^{l-l} B_{ii} \Delta u_i(k+l-1|k) + A_i^{l-1} B_{ij} \Delta u_{ni}(k|k) + \\ & + \dots + A_i^{l-l} B_{ij} \Delta u_{ni}(k+l-1|k) \\ & l = \overline{1, N_p} \end{aligned} \quad (\text{II.27})$$

$$\begin{aligned} y_i(k+l|k) = & C_i A_i^l x_i(k) + C_i A_i^{l-1} B_{ii} \Delta u_i(k|k) \\ & + \dots + C_i A_i^{l-l} B_{ii} \Delta u_i(k+l-1|k) + \\ & C_i A_i^{l-1} B_{ij} \Delta u_{ni}(k|k) + \dots + \\ & C_i A_i^{l-l} B_{ij} \Delta u_{ni}(k+l-1|k) \\ & l = \overline{1, N_p} \end{aligned} \quad (\text{II.28})$$

O remarcă importantă este faptul că funcția cost (II.25) depinde de starea măsurată la momentul curent $x_i(k)$ și de secvența de intrare viitoare a vecinului $\Delta U_{ni}(k)$ pentru că predictorul ieșirii Y_i , $i = \overline{1, N}$ este calculat în formă matricială pornind de la (II.28):

$$Y_i = \tilde{A}_i x_i(k) + \tilde{B}_{ii} \Delta U_i(k) + \tilde{B}_{ij} \Delta U_{ni}(k) \quad i = \overline{1, N}, j = i - 1 \quad (\text{II.29})$$

unde

$$\tilde{A}_i = [C_i A_i \quad C_i A_i^2 \quad \dots \quad C_i A_i^{N_p}]^T$$

$$\tilde{B}_{ii} = \begin{bmatrix} C_i B_{ii} & 0 & \dots & 0 \\ C_i A_i B_{ii} & C_i B_{ii} & \dots & 0 \\ \dots & \dots & \dots & \dots \\ C_i A_i^{N_p-1} B_{ii} & C_i A_i^{N_p-2} B_{ii} & \dots & C_i A_i^{N_p-N_c} B_{ii} \end{bmatrix}$$

și \tilde{B}_{ij} este obținut similar cu \tilde{B}_{ii} .

După substituția predictorului (II.29) în funcția cost (II.25) și calculele matriciale corespunzătoare, se obține următoarea formă explicită a funcției cost:

$$\begin{aligned} J_i(x_i(k), \Delta U_i, \Delta U_{ni}) = & (R_{si} - \tilde{A}_i x_i(k))^T (R_{si} - \tilde{A}_i x_i(k)) + 2 \Delta U_i^T \tilde{B}_{ii}^T \tilde{B}_{ij} \Delta U_{ni} \\ & - 2 \Delta U_i^T \tilde{B}_{ii}^T [R_{si} - \tilde{A}_i x_i(k)] - 2 \Delta U_{ni}^T \tilde{B}_{ij}^T [R_{si} - \tilde{A}_i x_i(k)] \\ & + 2 \Delta U_i^T (\tilde{B}_{ii}^T \tilde{B}_{ii} + R_i) \Delta U_i + \Delta U_{ni}^T (\tilde{B}_{ij}^T \tilde{B}_{ij}) \Delta U_{ni} \end{aligned} \quad (\text{II.30})$$

cu $i = \overline{1, N}$, $j = i - 1$.

După cum sa menționat mai devreme, în (II.30) variabila liberă este $\Delta U_i(k)$, $i = \overline{1, 2}$. Dacă nu sunt impuse restricții pentru variabila de control sau ieșiri, soluția optimală se obține egalând derivata de ordin unu a (II.30) cu zero, obținându-se:

$$\Delta U_i^*(k) = (\tilde{B}_{ii}^T \tilde{B}_{ii} + R_i)^{-1} \tilde{B}_{ii}^T [R_{si} - \tilde{A}_i x_i(k)] - (\tilde{B}_{ii}^T \tilde{B}_{ii} + R_i)^{-1} \tilde{B}_{ii}^T \tilde{B}_{ij} \Delta U_{ni}, \quad (\text{II.31})$$

cu $i = \overline{1, N}$, $j = i - 1$. Pe de altă parte, una din avantajele controlului predictiv este impunerea de restricții direct în funcția de optimizare. Pentru problema de contro specifică unei formații de vehicule lider-urmăritor, restricțiile sunt impuse pentru variația incrementală a variabilelor de control:

$$\Delta U_i(k) \leq \Delta U_{i-1}(k) \quad (\text{II.32})$$

care asigură faptul că, la fiecare perioadă de eșantionare, variația incrementală a accelerației fiecărui vehicul urmăritor este mai mică decât variația incrementală a vehiculului predecesor. În acest mod se asigură stabilitatea plutonului.

În acest raport, fiecare vehicul urmăritor trebuie să mențină constantă distanța dorită cu respect la vehiculul predecesor. Astfel, la fiecare moment de eșantionare, se măsoară poziția locală relativă la poziția predecesorului, din care se scade distanța dorită. Minimizând această eroare în zero, se asigură în mod implicit spațierea dorită între vehicule. În metoda DMPC non-cooperativă propusă, la fiecare moment de eșantionare, fiecare agent rezolvă o problemă de optimizare de dimensiuni reduse presupunând o traiectorie de intrare a vecinului fixată. De asemeni, se asumă un scenariu de comunicare fără întârzieri, în care la fiecare perioadă de eșantionare, fiecare vehicul urmăritor i primește traiectoria optimală de comandă de la vecinul său (i.e., vehiculul predecesor). Această traiectorie este prezisă pe întregul orizont de comandă, la fiecare eșantion de timp (i.e., $\Delta U_{ni}^*(k+l|k)$ cu $l = 0, N_c - 1$). Aici, indicele inferior ni definește predecesorul vehiculului i , care este controlat de agentul $i - 1$, $i = \overline{2, N}$. Vehiculul lider este controlat de agentul $i = 1$.

În metodologia propusă, regulatoarele distribuite sunt evaluate secvențial și doar o singură dată per perioadă de eșantionare. Astfel, la fiecare eșantion de timp, liderul plutonului își rezolvă problema de optimizare locală și trimite traiectoria optimală calculată către primul vehicul urmăritor. Acest vehicul urmăritor folosește traiectoria optimală primită, pentru a rezolva problema de optimizare proprie și trimite rezultatul următorului vehicul urmăritor din lanț. Cineva ar putea remarca că această arhitectură de control este validă atât timp cât perioada de eșantionare este suficient de lungă pentru a permite schimbul de informații și calculele aferente.

Algoritmul DMPC descris anterior poate fi rezumat astfel:

- Pas 1: *Inițializare*: stările $x_{pi}(k)$ sunt măsurate vectorul de stare augmentat $x_i(k)$ este calculat. Folosind soluția optimală de la momentul de timp k (i.e., $\Delta U_{ni}^*(k|k)$), traiectoria viitoare de intrare corespunzătoare vecinului $\Delta U_{ni}(k|k)$ este folosită în problema de optimizare;
- Pas 2: *Optimizare*:
 - Optimizare fără restricții: Fie problema de control optimală:

$$\min_{\Delta U_i(k)} J_i$$

cu J_i definit în ecuația (II.30), sau

– optimizare cu restricții: Fie problema de control optimală:

$$\min_{\Delta U_i(k)} J_i$$

cu J_i definit în ecuația (II.30) și restricțiile (II.32).

- Pas 3: *Implementare*: se trimite către sub-sistem semnalul de intrare care este primul element din soluția optimală integrată (i.e., $u(k) = u(k-1) + \Delta u_i^*(k|k)$)
- Pas 4: *Comunicare*: se trimite către agentul succesori, soluția optimală calculată la momentul de timp k (i.e., $\Delta U_i^*(k)$), se incrementează variabila de timp la $k+1$ și se revine la pasul 1 la următorul eșantion de timp.

În concluzie, fiecare agent $i, i = \overline{1, N}$ rezolvă o problemă de optimizare locală folosind: *i*) starea măsurată la momentul curent $x_{pi}(k)$ și *ii*) strategia de comandă optimală primită de la agentul vecin cuplat.

Rezultate obținute În continuare, se vor prezenta rezultatele obținute simulând un ploton de $N = 5$ vehicule identice modelate ca dublu integrator. Folosind o perioadă de eșantionare de 1 secundă, modelul fiecărui vehicul următor $i, i = \overline{2, N}$ este obținut din (II.20), în timp ce modelul liderului se obține din (II.38).

Algoritmul DMPC non-cooperativ implementat a fost simulat într-un experiment cu urmărirea unei referințe rampă structurat după cum urmează: referința de poziție pentru vehiculul lider a fost alcătuită dintr-o succesiune de trei rampe având schimbarea de pantă la eșantioanele de timp 10s și 20s. Obiectivul de control pentru lider a fost impus ca urmărirea referinței fără eroare staționară, în timp ce obiectivele vehiculelor următor au fost să se minimizeze stările în zero, care implică o spațiere constantă între vehiculele urmaritor.

În figura II.2, rezultatele problemei de optimizare fără restricții sunt marcate cu linie continuă albastră, în timp ce cu linie roșie punctată sunt date rezultatele problemei de optimizare cu restricțiile definite în (II.32). În prima linie sunt figurate poziția absolută, viteza și accelerația liderului, iar în liniile următoare, sunt marcate eroarea relativă de poziție, de viteză și accelerația vehiculelor următor. Timpul în secunde este dat pe axa orizontală. Să se noteze că formularea cu restricțiile (II.32) limitează variația incrementală a variabilei de control corespunzătoare vehiculului curent, la valoarea vehiculului predecesor. Astfel, problema de optimizare a liderului este fără restricții deoarece liderul nu are predecesor. Aceasta se observă în primul rând din figură deoarece linia roșie și albastră coincid. Următorii parametri de optimizare $N_p = 20$, $N_c = 10$ au fost folosiți în algoritmul DMPC. Factorul de ponderare a intrării $\alpha = 0.1$ s-a folosit pentru toate vehiculele.

În figura II.3, sunt detaliate rezultatele liderului pentru urmărirea referinței (i.e., acestea au fost afișate și în figura II.2, primul rând, fereastra din stânga). Rezultatele arată că liderul ajunge în regim staționar în 3s, 4s și 5s, raportat la a doua și a treia schimbare de pantă în traiectoria de referință.

Pentru a evalua performanța obținută în cele două scenarii, s-a calculat indicele de performanță eroare medie pătratică MSE (eng. Mean Squared Error):

$$MSE_i = \frac{1}{N_r} \sum_{p=1}^{N_r} (r_i(p) - y_i(p))^2 \quad (II.33)$$

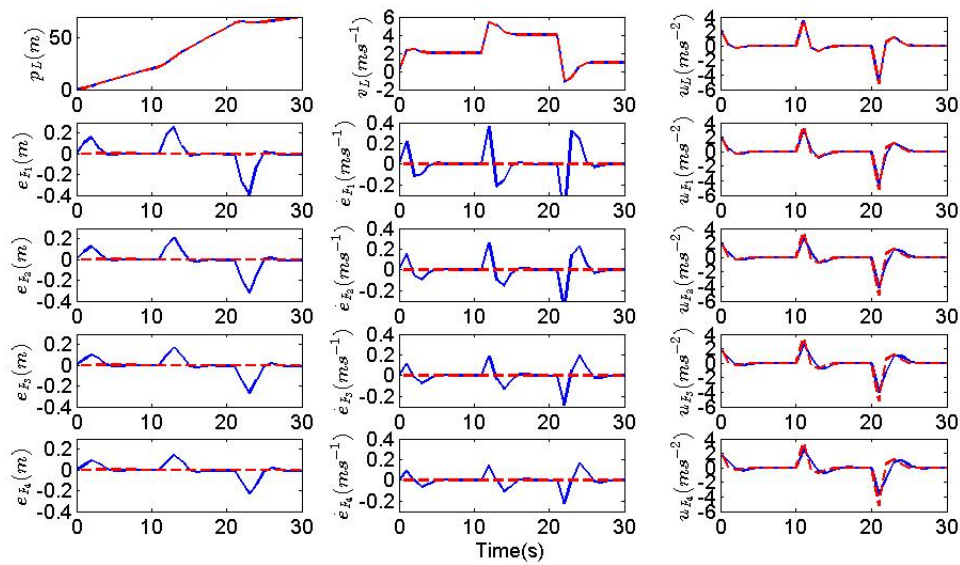


Figura II.2: Rezultatele comparative între soluția problemei fără restricții marcată cu linie albastră continuă și soluția cu restricții afișată cu linie roșie continuă, simulate cu un pluton de $N=5$ vehicule.

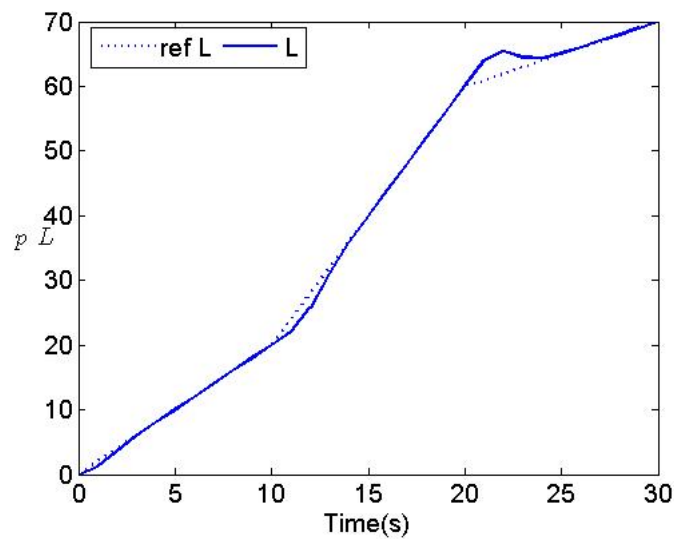


Figura II.3: Rezultatele obținute pentru lider în experimentul de urmărirea referinței.

Table 1: Valorile MSE obținute în cele două scenarii

| $MSE(m^2)$ | Test 1 | Test 2 |
|------------|--------|--------|
| L | 1.1428 | 1.1428 |
| F1 | 0.0130 | 0 |
| F2 | 0.0084 | 0 |
| F3 | 0.0059 | 0 |
| F4 | 0.0044 | 0 |

unde $i = 1$ este indicele vehiculului lider și Nr este numărul total de eșantioane evaluate (i.e., de la momentul de timp $0s$ până la $30s$). Cu $r_i(p)$ și $y_i(p)$, $p = \overline{1, Nr}$ s-au notat referința impusă și ieșirea măsurată (i.e., poziția) la fiecare eșantion de timp. Valorile indicelui MSE sunt date în tabelul 3, unde Test 1 și 2 semnifică cazul fără restricții, respectiv cu restricții.

Conform indicelui MSE , se poate remarca faptul că liderul are același indice de performanță în ambele cazuri, deoarece ambele experimente sunt fără restricții impuse pentru lider. În cazul fără restricții, indicele MSE descrește spre capătul plutonului, în timp ce în cazul cu restricții, toate vehiculele următor au eroare zero. Acest rezultat era de așteptat luând în considerație faptul că, în strategia DMPC vehiculele următor primesc soluția optimă de la predecesor, la același eșantion de timp îi folosesc această informație pentru a calcula comanda optimă locală. Astfel, în Testul 1, plutonul se stabilizează natural fără restricții. Totuși, scopul studiului a fost obținerea celor mai bune rezultate posibile, care se obțin când se impun restricții în conformitate cu dinamica plutonului. În cazul vehiculelor următor, rezultatele date în figura II.2 corespund cu indicele MSE calculat, în care scenariul cu restricții are performanțe mai bune decât cazul nominal. Aceasta se interpretează prin faptul că vehiculele următor sunt conduse cu aceeași viteză ca ‘liderii relativi’ corespunzători, și mențin o spațiere constantă pe întreg experimentul.

Ca și concluzii, în cadrul acestui raport, în acest capitol, s-a prezentat conceptul unui algoritm DMPC potrivit pentru controlul unui pluton de vehicule. Algoritmul a fost testat în două cazuri experimentale, ambele presupunând că comunicarea între vehicule este fără întârzieri (i.e., comunicarea între două vehicule consecutive și rezolvarea problemei de optimizare au loc în același eșantion de timp). În cazul nominal, plutonul este condus rezolvând probleme de optimizare fără restricții. În cel de-al doilea caz, se impun restricții pentru variația incrementală a variabilei de control și rezultatele arată o îmbunătățire a performanțelor.

În continuare se va analiza rezultatele obținute pentru modele de vehicule mai complexe. De asemenea, se va aborda un scenariu de comunicare cu întârzieri (i.e., problema de optimizare este rezolvată local folosind informații recepționate de la predecesor cu o anumită întârziere).

II.2.1.5 Pluton de vehicule descrise prin model realist

În această secțiune se va generaliza algoritmului non-cooperativ DMPC descris anterior și testat pe un pluton de vehicule modelate cu dublu integrator, într-o arhitectură de control mai complexă, în care vehiculele sunt modelate realist printr-o descriere a lanțului de

transmisie a puterii într-un vehicul [135].

Modelarea vehiculelor prin descrierea lanțului de transmisie a puterii Lanțul de transmisie a puterii într-un vehicul este un sistem mecanic care transmite puterea motorului către roți. Acesta poate fi modelat în spațiul stărilor de forma:

$$\dot{x}^m(t) = A_c^m x^m(t) + b_c^m u^m(t) + f_c^m, \quad t \in \mathbb{R}_+ \quad (\text{II.34})$$

unde $x^m(t) = (x_1^m(t) \ x_2^m(t) \ x_3^m(t) \ x_4^m(t))^T \in \mathbb{R}^4$ sunt stările sistemului (i.e., viteza motorului, viteza roții, accelerația unghiulară și poziția vehiculului), $u^m(t) \in \mathbb{R}$ este intrarea sistemului (cuplul motor), $A_c^m \in \mathbb{R}^{4 \times 4}$ and $b_c^m \in \mathbb{R}^{4 \times 1}$ sunt matricile sistemului și $f_c^m \in \mathbb{R}^{4 \times 1}$ este un termen afin:

$$A_c^m = \begin{pmatrix} -\frac{b_e}{J_{eg}} - \frac{b_f}{i_{tot}^2 J_{eg}} & \frac{b_f}{i_{tot} J_{eg}} & -\frac{k_f}{i_{tot} J_{eg}} & 0 \\ \frac{b_f}{i_{tot} J_v} & -\frac{b_a + b_f}{J_v} & \frac{k_f}{J_v} & 0 \\ \frac{1}{i_{tot}} & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{pmatrix},$$

$$b_c^m = \begin{pmatrix} \frac{1}{J_{eg}} \\ 0 \\ 0 \\ 0 \end{pmatrix}, \quad f_c^m = \begin{pmatrix} 0 \\ -\frac{T_{roll} + T_{grade}}{J_v} \\ 0 \\ 0 \end{pmatrix}.$$

Astfel, fiecare vehicul din pluton este descris de modelul (II.163). Obiectivul de control pentru vehiculul lider este minimizarea unei referințe de viteză iar obiectivul vehiculelor următor este păstrarea unei distanțe constante raportată la poziția măsurată a vehiculului predecesor. Pentru a implementa legea de control a plutonului, trebuie definită relația dintre vehicul și predecesorul său [136]:

$$h_i(t) = x_{i+1}(t) - x_i(t) - \dot{h}_i(t) = v_{i+1}(t) - v_i(t) \quad (\text{II.35})$$

$$\dot{v}_i(t) = \alpha(v_i^{ref}(t) - v_i(t)) + \beta(v_{i+1}(t) - v_i(t)) \quad (\text{II.36})$$

unde l este lungimea vehiculului i , cu $i = \overline{1, N}$, $x_{i+1}(t)$ este poziția vehiculului predecesor $i + 1$ și $x_i(t)$ este poziția vehiculului curent i . Cu v_{i+1} și v_i s-au notat vitezele vehiculului precedent respectiv vehiculului curent. Accelerația vehiculelor este dată de legea (II.36), în care primul termen ponderat de constanta α , minimizează eroarea de viteză a vehiculului curent și cel de-al doilea termen ponderat de β minimizează diferența de viteză între vehiculul curent și predecesorul său. Astfel, din punct de vedere distribuit, este necesară comunicarea vitezei fiecărui vehicul către succesul său, pentru a putea implementa legea de urmărire lider-următor (II.36).

Rezultatele obținute în simulare Algoritm DMPC non-cooperativ descris anterior a fost adaptat la noul model al vehiculului (II.163) și testat într-un experiment cu perturbații de tip întârzieri în transmisia semnalelor între două vehicule consecutive. Astfel, pentru un

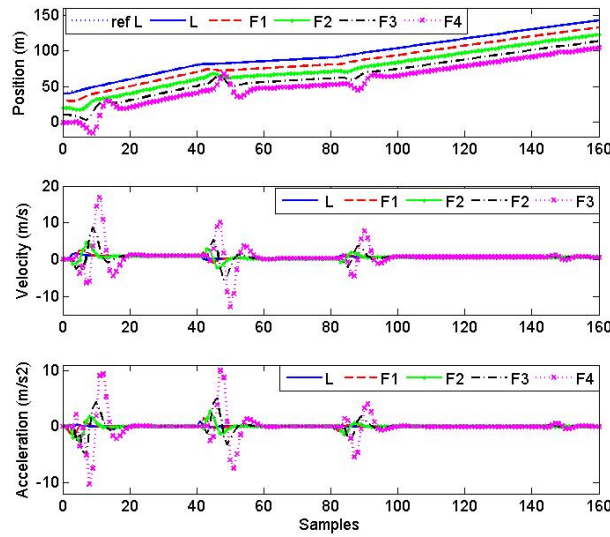


Figura II.4: Rezultat obținut în simularea unui pluton de $N=5$ vehicule, într-un test cu întârzieri de comunicare de 1 eșantion.

pluton de $N = 5$ vehicule s-a implementat un experiment cu întârzieri de un eșantion în transmisia comenzii optimale. Deoarece, întârzierea semnalului este cunoscută, s-au luat măsuri pentru a îmbunătăți performanța, prin șiftarea vectorului ce comandă optimală primită de la predecesor, cu valoarea întârzierii (i.e., un eșantion) și dublarea ultimului element din vector, pentru a păstra aceeași lungime a vectorilor. În Figura II.4 este dat rezultatul obținut în simulare.

În acest experiment cu întârzieri de comunicare se observă o degradare a performanței din punct de vedere a stabilității plutonului și a efortului de control. Acestea sunt datorate erorilor care se propagă în sistem produse de diferența dintre traiectoria optimală și șiftată folosită în problema de optimizare și traiectoria optimală reală calculată de predecesor și transmisă către succesori cu întârziere.

Ca și concluzie, algoritmul non-cooperativ DMPC cu modelarea vehiculelor prin dinamica lanțului de transmisie a puterii a fost testat într-un experiment în care s-au introdus perturbații de comunicare. Rezultatul obținut arată o creștere a efortului de control pe măsură ce numărul de vehicule din pluton crește. În continuare, se vor analiza alte modalități de a combate efectele negative ale perturbațiilor din proces.

II.2.2 Controlul vitezei de croazieră și a distanței dintre vehiculele unui pluton folosind un algoritm de control predictiv distribuit

În această secțiune se vor implementa două strategii de control predictiv distribuit în funcție de poziția vehiculului în cadrul plutonului. Astfel, pentru vehiculul lider poziționat în capătul coloanei, este controlată viteza raportată la viteza de croazieră dorită de către șofer, urmând ca pentru celelalte vehicule urmăriitoare să fie controlată distanța față de vehiculul precedent. Fiecare vehicul din pluton este integrat într-o arhitectură de control predictiv

distribuit cu restricții pe intrări și restricții terminale definite pentru vehiculele următoare asigurând stabilitatea strategiei de control. Rezultatele propuse sunt disponibile în [137]

II.2.2.1 Introducere

Două dintre cele mai comune strategii de control în industria automotivă sunt controlul vitezei șde croazieră și menținerea unei distanțe constante în vehicule. Controlul vitezei șde croazieră înseamnă asigurarea în mod automat, a unei viteze longitudinale constante egală cu viteza dorită de că tre șofer, în timp ce controlul distanței menține vehiculul la o distanță dorită , sigură față de vehiculul din față [138].

Dezvoltarea unei strategii de control pentru plutoane de vehicule prin combinarea acestor două strategii de control a dobândit în ultimii ani un interes crescut în comunitatea de control [14, 20, 139], datorită potențialului în asigurarea: siguranței șoselelor, creșterii capacită ții autostă zilor, confortului pasagerilor și eficienței economice. Acestea sunt obținute prin conducerea vehiculelor în cadrul unui pluton cu o viteză constantă specificată , menținând o distanță de siguranță între vehicule [22, 140].

Pornind de la particularită țile specifice plutoanelor de vehicule (i.e., vehiculele fiind cuplate într-o arhitectură în lanț), strategiile de control predictiv distribuit DMPC (eng. Distributed Model Predictive Control) pot fi folosite la implementarea controlului. Astfel, sistemul (adică plutonul) este descompus în mai multe sub-sisteme (sau vehicule) cuplate prin intrări, stări sau restricții, fiecare fiind controlat de un regulator individual (sau agent). Fiecare agent rezolvă o problemă de optimizare locală , folosind atât mă sură tori locale cât și informații primite de la agenții vecini prin intermediul rețelei de comunicare [141].

În [142] este propus un algoritm DMPC pentru controlul plutoanelor de vehicule heterogene cuplate prin restricții. Stabilitatea plutonului este obținută folosind o restricție terminală care asigură că pentru fiecare vehicul, starea terminală la finalul orizontului de predicție converge la starea dorită , folosind o topologie de comunicare unidirecțională . Problema controlului unei formații de vehicule care urmă rește șoseaua este definită în [143]. Fiecare vehicul rezolvă o problemă de urmărire locală , bazată pe traiectoria liderului și structura predefinită a formației, folosind informații obținute de la vecini. Totodată , este folosită o abordare hibridă care combină o problemă de optimizare predictivă MPC (eng. Model Predictive Control) cu reguli logice, simplificând descrierea restricțiilor de ocolire a obstacolelor in problema de oprimizare. În [144] este prezentat un algoritm DMPC pentru controlul unei formații alcă tuite din mai mulți agenți cu restricții definite pentru evitarea coliziunilor între vehicule și restricții pentru evitarea obstacolelor din traseu. Aceste restricții sunt satisfă cute prin utilizarea unui set invariant pentru restricția terminală , o lege de reglare terminală si o funcție de cost terminală . În [145] este descrisă o arhitectură de control cu două nivele aplicată pentru plutoane de camioane. Nivelul superior este alcă tuit dintr-un coordonator de pluton care calculează o traiectorie de viteza eficientă d.p.d.v. a carburantului folosind informații cu privire la topografia traseului. Acest profil de viteză este transmis că tre nivelul inferior unde este folosit ca referință pentru regulatoarele DMPC.

O primă implementare a unei strategii de control distribuit DMPC pentru un pluton de vehicule a fost prezentată în [146]. Pornind de la rezultatele din [30, 123] descrise pentru doi agenți, algoritmul de control a fost reformulat pentru agenți multipli într-o arhitec-

tură în lanț, caracterizată printr-o structură de comunicare și cuplare unidirecțională (i.e., informația este transmisă de la lider către succesorul său). Rezultatele fără restricții au fost comparate cu cele obținute folosind restricții definite ca rata de variație a intrării să fie mai mică decât variația intrării vehiculului predecesor.

În acest capitol, este propus un algoritm de control predictiv distribuit DMPC pentru controlul vitezei de croazieră și distanței intra-vehiculare, potrivit pentru reglarea unui pluton de vehicule. Deoarece liderul nu are alt vehicul în față, pentru acesta va fi controlată doar viteza, urmând ca pentru vehiculele urmăritoare să fie reglată distanța față de predecesor. În structura de control lider-urmăritor propusă, fiecare vehicul este considerat un lider relativ pentru vehiculul succesor. Stabilitatea plutonului este obținută prin optimizare cu restricții, în care valoarea maximă admisibilă pentru intrarea liderului a fost impusă și pentru restul plutonului. De asemenea, s-a definit o restricție terminală pentru valoarea absolută a erorii de distanță, asigurând că fiecare vehicul urmăritor are eroarea de distanță în modul mai mică sau egală ca valoarea vehiculului precedent.

Principalele diferențe față de rezultatele anterioare propuse în [146] sunt:

- folosirea unui model realist pentru vehicul (II.37) în locul reprezentării ca dublu integrator.
- definirea unei funcții de cost cu o componentă terminală (II.44), cu o restricție terminală în care pentru fiecare urmăritor, modulul erorii de distanță este mai mic decât valoarea vehiculului precedent.
- folosirea unui model extins (II.41), (II.42) cu o componentă integrală pentru eroarea de control care să asigure urmărirea referinței, înlocuind descrierea cu model de viteză

II.2.2.2 Modelarea vehiculelor și formularea problemei de control

Fie un pluton alcătuit din N vehicule modelate pornind de la ecuația de mișcare longitudinală care caracterizează dinamica unui vehicul (i.e., care analizează dinamica unui vehicul în termeni de accelerație și decelerație), aplicată pe o diagramă de corp-liber descrisă în [138].

Ecuția mișcării liniarizată este:

$$\dot{v} = \frac{1}{\tau}v + \frac{K}{\tau}F \quad (\text{II.37})$$

unde v este viteza de înaintare și F este forța de tracțiune.

Constanta de timp τ și factorul de amplificare al sistemului K sunt definite ca:

$$\tau = \frac{m}{\rho C_d A (v_0 + v_w)}$$

$$K = \frac{1}{\rho C_d A (v_0 + v_w)}$$

folosind următorii parametri ai vehiculului: masa vehiculului m , densitatea aerului ρ , coeficientul de frecare C_d , aria secțiunii vehiculului A , viteza vântului v_w și valoarea de echilibru a vitezei v_0 folosită în liniarizare.

Modelul simplu dat în (II.37) asumă că dinamica elementului de execuție (care manipulând unghiul pedalei de accelerație are ca ieșire forța de tracțiune) este neglijabilă și că constanta de timp τ și factorul de amplificare K al modelului vehiculului sunt constante. Aceste deficiențe sunt tratate în lucrări ca [147, 148]. Prima ipoteză nu este improbabilă dacă se consideră faptul că constanta de timp a elementului de execuție este mult mai mică decât constanta de timp a vehiculului, putând fi astfel neglijată [149].

În cazul controlului unui pluton, vehiculul lider este controlat de agentul $i = 1$. Deoarece primul vehicul nu are un alt vehicul în față, strategia de control se reduce doar la reglarea vitezei de croazieră. Pornind de la modelul dat în (II.37), este derivat modelul corespunzător în spațiul stărilor:

$$\begin{aligned} \dot{x}_1 &= \begin{bmatrix} 0 & 1 \\ 0 & \frac{-1}{\tau_1} \end{bmatrix} x_1 + \begin{bmatrix} 0 \\ \frac{K_1}{\tau_1} \end{bmatrix} u_1 \\ y_1 &= \begin{bmatrix} 0 & 1 \end{bmatrix} x_1 \end{aligned} \quad (\text{II.38})$$

unde stările x_1 sunt respectiv, poziția vehiculului p_1 , și viteza de înaintare v_1 ; mărimea de intrare u_1 este forța de tracțiune F_1 , iar ieșirea y_1 este viteza vehiculului v_1 . Pentru simplitate, doar modelul continuu este definit, care este discretizat pentru elaborarea strategiei de control.

Folosind informații masurate local, agentul corespunzător vehiculului lider rezolvă o problemă de optimizare pentru reglarea vitezei de croazieră, asigurând urmărirea referinței de viteză r_1 fără eroare staționară. Întrucât plutonul are o arhitectură în lanț, agentul trebuie să comunice traiectoria de comandă, valoarea curentă a vitezei și parametri vehiculului doar către succesorul său (i.e., controlat de agentul $i = 2$).

Modelele vehiculelor urmăritoare sunt definite ținând cont de faptul că în interiorul plutonului, fiecare agent i , $i = \overline{2, N}$ responsabil pentru vehiculul corespunzător, trebuie să asigure o strategie de control a distanței între vehicule (adică să mențină vehiculul la o distanță de siguranță, impusă raportat la vehiculul din față).

Vehiculele urmăritoare $i \in \{2 \dots N\}$ sunt caracterizate de:

$$\begin{aligned} \dot{x}_i &= \begin{bmatrix} 0 & -1 & 1 \\ 0 & \frac{-1}{\tau_i} & 0 \\ 0 & 0 & \frac{-1}{\tau_{i-1}} \end{bmatrix} x_i \\ &+ \begin{bmatrix} 0 \\ \frac{K_i}{\tau_i} \\ 0 \end{bmatrix} u_i + \begin{bmatrix} 0 \\ 0 \\ \frac{K_{i-1}}{\tau_{i-1}} \end{bmatrix} u_{i-1} \\ y_i &= \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} x_i \end{aligned} \quad (\text{II.39})$$

unde stările x_i , $i = \overline{2, N}$ sunt distanța d_i măsurată de agentul i între vehiculul următor i și liderul său relativ (vehiculul $i - 1$), viteza de înaintare v_i , și viteza de înaintare a liderului relativ v_{i-1} ; mărimea de comandă u_i este forța de tracțiune, iar u_{i-1} este mărimea de comandă transmisă de către predecesor prin intermediul rețelei de comunicare, ieșirea este distanța măsurată față de predecesor d_i .

Fiecare agent responsabil pentru un vehicul următor i , $i = \overline{2, N}$ are o legătură de comunicare bidirecțională. Pentru a putea calcula soluția locală, are nevoie de întraga

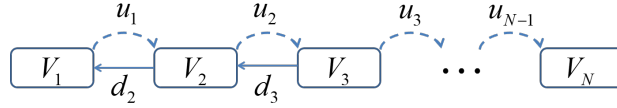


Figura II.5: Reprezentație conceptuală a unui pluton cu N vehicule.

secvență predictată de intrări, și de valoarea vitezei transmise de către vehiculul predecesor $i - 1$. După ce problema de optimizare DMPC este minimizată, traiectoria predictată a intrării și viteza măsurată este transmisă către succesori (cu indicele $i + 1$). Notați că, plutonul se termină cu vehiculul cu indicele $i = N$ care are o legătură de comunicare unidirecțională doar cu vehiculul predecesor.

O altă remarcă este faptul că modelul descris în (II.39) depinde de parametri vehiculului precedent. Întrucât în acest capitol se presupune că plutonul este omogen (i.e., alcătuit din vehicule identice), toți parametri vehiculelor sunt egali (adică $K_{i-1} = K_i$ și $\tau_{i-1} = \tau_i$). Cu toate acestea, pentru un pluton heterogen, în care toate vehiculele sunt diferite, strategia de control DMPC permite comunicarea între vehiculele consecutive a parametrilor acestora la fiecare perioadă de eșantionare sau o singură dată în etapa de initializare când plutonul este configurat. Pentru fiecare vehicul următor, în problema de optimizare se impune o referință pentru distanța măsurată față de vehiculul din față.

În figura II.5 este reprezentată o diagramă conceptuală a unui pluton cu N vehicule în arhitectură în lanț. Distanța dintre două vehicule consecutive este marcată cu d_i , $i = \overline{2, N}$ măsurată de fiecare agent responsabil cu un vehicul următor V_i , $i = \overline{2, N}$. Informația transmisă între vehicule este marcată cu săgeți punctate (i.e., variabila de control u_i , $i = \overline{1, N - 1}$).

II.2.2.3 Implementarea algoritmului DMPC

Fie următoarea definiție pentru dinamica vehiculelor următoare, cu particularitatea de a fi cuplate printr-o arhitectură în lanț (adică sub-sistemul i , $i = \overline{2, N}$ este cuplat doar cu vehiculul predecesor, notat cu indicele $i - 1$):

$$\begin{aligned} x_i(k+1) &= A_i x_i(k) + B_{i,i} u_i(k) + B_{i,i-1} u_{i-1}(k) \\ y_i(k) &= C_i x_i(k) \end{aligned} \quad (\text{II.40})$$

unde k este timpul discret, $u_i \in R$, $y_i \in R$ and $x_i \in R^3$ sunt variabilele de intrare, ieșire și stare ale sub-sistemului i , cu $i = \overline{2, N}$.

În timp ce în [146], un integrator a fost introdus în sistem prin intermediul unui model de viteză, având vectorul de stare augmentat cu ieșirea și incrementele stării, în această secțiune modelul este extins cu o componentă integrală pe eroarea de control $\bar{x}_i(k+1) =$

$\bar{x}_i(k) - C_i x_i(k) + r_i(k)$ după cum urmează:

$$\begin{aligned} \begin{bmatrix} x_i(k+1) \\ \bar{x}_i(k+1) \end{bmatrix} &= \begin{bmatrix} A_i & O \\ -C_i & I \end{bmatrix} \begin{bmatrix} x_i(k) \\ \bar{x}_i(k) \end{bmatrix} + \begin{bmatrix} O \\ I \end{bmatrix} r_i(k) \\ &+ \begin{bmatrix} B_{i,i} \\ O \end{bmatrix} u_i(k) + \begin{bmatrix} B_{i,i-1} \\ O \end{bmatrix} u_{i-1}(k) \\ y_i(k) &= [C_i \quad O] \begin{bmatrix} x_i(k) \\ \bar{x}_i(k) \end{bmatrix} \end{aligned} \quad (\text{II.41})$$

unde I și O notează respectiv matricea identitate și nulă, cu dimensiuni adecvate.

Deoarece liderul plutonului nu are nici un alt vehicul în față, modelul acestuia poate fi obținut din (II.40), respectiv (II.41) pentru indicele $i = 1$, prin neglijarea termenului dependent de intrarea vecinului, obținându-se următorul model extins:

$$\begin{aligned} \begin{bmatrix} x_i(k+1) \\ \bar{x}_i(k+1) \end{bmatrix} &= \begin{bmatrix} A_i & O \\ -C_i & I \end{bmatrix} \begin{bmatrix} x_i(k) \\ \bar{x}_i(k) \end{bmatrix} \\ &+ \begin{bmatrix} O \\ I \end{bmatrix} r_i(k) + \begin{bmatrix} B_{i,i} \\ O \end{bmatrix} u_i(k) \\ y_i(k) &= [C_i \quad O] \begin{bmatrix} x_i(k) \\ \bar{x}_i(k) \end{bmatrix} \end{aligned} \quad (\text{II.42})$$

Pentru toate sub-sistemele, s-au considerat următoarele restricții pe intrări:

$$u_i \in \mathcal{U}_i, \quad i = \overline{1, N} \quad (\text{II.43})$$

unde setul \mathcal{U}_i , $i = \overline{1, N}$ este definit de setul de inegalități din (II.46).

În algoritmul DMPC propus, obiectivul de control pentru fiecare sub-sistem este definit în funcție de poziția sa în cadrul plutonului, după cum urmează:

- pentru vehiculul lider al plutonului se formulează o problemă de control a vitezei de croazieră ținând cont de modelul din (II.42) unde ieșirea este viteza vehicului. Reglarea este obținută prin impunerea în cadrul problemei de optimizare a unei referințe constante de-a lungul orizontului de predicție, egală cu viteza dorită $\tilde{v}_i(k)$ la momentul de eșantionare k :

$$r_i(k+l|k) = \tilde{v}_i(k), \quad l = 1, \dots, N_p, \quad i = 1$$

- pentru vehiculele urmăritoare se formulează o problemă de control a distanței raportate la vehiculul precedent, folosind modelul din (II.41). Aceasta este rezolvată prin impunerea în cadrul problemei de optimizare a unei referințe constante egală cu distanța dorită între două vehicule consecutive \tilde{d}_i . Astfel se asigură ca vehiculele urmăritoare pastrează o distanță de siguranță față de vehiculul din față.

$$r_i(k+l|k) = \tilde{d}_i(k), \quad l = 1, \dots, N_p, \quad i = \overline{2, N}$$

Pentru vehiculele urmăritoare, problema de optimizare este definită ca $i = \overline{2, N}$:

$$\begin{aligned}
J_i(x_i^0) &= \min_{u_i^0, \dots, u_i^{N_p-1}} \sum_{l=1}^{N_p-1} \|r_i^l - y_i^l\|_2 + \sum_{l=0}^{N_p-1} \|R_{u_i} u_i^l\|_2 \\
&\quad + \|r_i^{N_p} - y_i^{N_p}\|_2 \\
\text{subj. to} &\quad (II.41) \\
&\quad \forall u_i^l \in \mathcal{U}_i, \quad l = 0, \dots, N_p - 1 \\
&\quad |r_i^{N_p} - y_i^{N_p}| \leq |r_{i-1}^{N_p} - y_{i-1}^{N_p}|
\end{aligned} \tag{II.44}$$

în timp ce liderul $i = 1$ optimizează:

$$\begin{aligned}
J_i(x_i^0) &= \min_{u_i^0, \dots, u_i^{N_p-1}} \sum_{l=0}^{N_p-1} \|r_i^{l+1} - y_i^{l+1}\|_2 + \|R_{u_i} u_i^l\|_2 \\
\text{subj. to} &\quad (II.42) \\
&\quad \forall u_i^l \in \mathcal{U}_i \\
&\quad \text{for } l = 0, \dots, N_p - 1
\end{aligned} \tag{II.45}$$

unde $y_i^l = y_i(k + l|k)$ notează pentru sub-sistemul i , valorile viitoare ale ieșirii, predictate de-a lungul orizontului de timp l , postulat la timpul k , având orizontul de predicție N_p ; pentru vehiculele urmăritoare, această secvență este obținută pornind de la starea inițială $x_i^0 = x_i(k)$ măsurată la timpul k , și aplicând modelului (II.41) secvența de intrări locale $\{u_i^0, \dots, u_i^{N_p-1}\}$ și secvența de intrări primite de la vehiculul predecesor $\{u_{i-1}^0, \dots, u_{i-1}^{N_p-1}\}$; $y_i^{N_p}$ este ieșirea viitoare la finalul orizontului de predicție, calculată pentru ieșirea (adică viteza de înaintare) sub-sistemului i , $i = \overline{2, N}$; r_i^l este vectorul de referință a ieșirii, impus de*-a lungul orizontului de predicție; $\|R_{u_i} u_i^l\|_2$ notează norma-2 în R și R_{u_i} este o matrice nesingulară; $|r_i^{N_p} - y_i^{N_p}|$ este valoarea absolută pentru eroarea de distanță la finalul orizontului de predicție;

Pentru toate vehiculele urmăritoare, funcția de cost definită în (II.44) are două componente. Primul termen minimizează eroarea predictată de-a lungul orizontului de predicție, în timp ce al doilea termen minimizează valoarea comenzii u_i . Componenta terminală penalizează eroarea predictată la finalul orizontului de predicție. Problema de optimizare a liderului (II.45) este definită fără componenta terminală.

Este important de menționat că în algoritmul propus, doar problema de optimizare a vehiculelor urmăritoare (II.44) are restricții terminale. Aceasta garantează ca la finalul orizontului de predicție, eroarea predictată de distanță în modul este mai mică sau egală cu valoarea corespunzătoare a liderului relativ, asigurându-se astfel stabilitatea plutonului (adica eroarea de distanță dintre vehicule nu crește). De-asemena, pentru ambele tipuri de vehicule, lider și urmăritori, sunt definite restricții pe intrări (II.51) definite în explicit:

$$u_i^{\min} \leq u_i^l \leq u_i^{\max}, \quad l = 0, \dots, N_p - 1 \tag{II.46}$$

unde limitele minime și maxime, respectiv u_i^{\min} și u_i^{\max} , $i = \overline{1, N}$ sunt ne-negative.

În această secțiune este implementat un algoritm DMPC secvențial și non-iterativ. Într-o perioadă de eșantionare, agentul responsabil cu reglarea liderului, calculează local soluția sa optimă și transmite informațiile relevante către vehiculul următor consecutiv. După ce datele sunt recepționate prin rețeaua de comunicare și măsurătorile locale sunt achiziționate, agentul responsabil cu al doilea vehicul din pluton minimizează problema de optimizare locală și transferă soluția locală către vehiculul succesiv.

II.2.2.4 Rezultate obținute prin simulare

În această secțiune sunt prezentate rezultatele obținute în simulare pentru un pluton omogen alcătuit din $N = 5$ vehicule. Valorile numerice ale parametrilor vehiculului $m = 1000kg$, $\rho = 1.202kg/m^3$, $C_d = 0.5$, $A = 1.5m^2$, $v_w = 0m/s$ și $v_0 = 20m/s$ au fost selectate conform specificațiilor din [138]. Modelul discret pentru lider s-a obținut folosind o perioadă de eșantionare de $T_s = 1$ secundă și funcția Matlab *c2dm*

$$\begin{aligned} x_1(k+1) &= \begin{bmatrix} 1 & 0.991 \\ 0 & 0.982 \end{bmatrix} x_1(k) \\ &+ \begin{bmatrix} 0.0005 \\ 0.0010 \end{bmatrix} u_1(k) \\ y_1(k) &= \begin{bmatrix} 0 & 1 \end{bmatrix} x_1(k) \end{aligned}$$

respectiv modelul vehiculului următor

$$\begin{aligned} x_i(k) &= \begin{bmatrix} 1 & -0.991 & 0.991 \\ 0 & 0.982 & 0 \\ 0 & 0 & 0.982 \end{bmatrix} x_i(k) \\ &+ \begin{bmatrix} 0.0005 \\ 0.0010 \\ 0 \end{bmatrix} u_i(k) + \begin{bmatrix} 0.0005 \\ 0 \\ 0.0010 \end{bmatrix} u_{i-1}(k) \\ y_i &= \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} x_i(k) \end{aligned}$$

Pentru control pot fi derivate ușor și modelele discrete extinse corespunzătoare liderului și următorilor, pornind de la formulările din (II.42), și respectiv (II.41).

Toate sub-sistemele i , $i = \overline{1, N}$ au următoarele restricții pe intrări:

$$\begin{cases} u_i^{\min} = 0 \\ u_i^{\max} = 3000 \end{cases} \quad (\text{II.47})$$

Testele în simulare au fost efectuate pe un calculator de 2.20 GHz, cu 4 GB memorie RAM, în Matlab 2014, folosind toolbox-ul YALMIP pentru modelare și optimizare [150], cu orizontul de predicție $N_p = 15$ eșantioane. Au fost testate două cazuri de optimizare având ponderile matricilor de intrare $R_{u_i} = 0.001$ și $R_{u_i} = 0.0005$.

Rezultatele în simulare au fost obținute într-un test pentru urmărirea referinței definit astfel: valoarea inițială pentru traiectoria de viteză a liderului a fost selectată în jurul valorii de echilibru $20m/s$ și au fost efectuate câteva schimbări de referință după cum urmează:

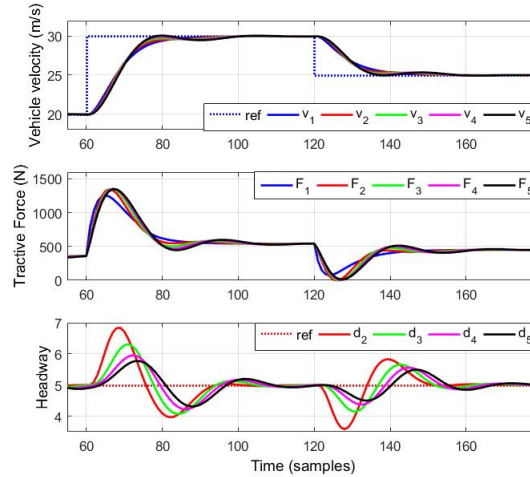


Figura II.6: Rezultate în simulare obținute pentru intrarea ponderată cu valoarea $R_{u_i} = 0.001$.

la momentul de timp 60s referința a fost schimbată din $20m/s$ în $30m/s$, urmată de altă schimbare la momentul 120s din $30m/s$ în $25m/s$. Valoarea dorită pentru distanța dintre vehicule a fost aleasă $\tilde{d}_i = 5$ m, $i = \overline{2, 4}$.

Rezultatele prezentate în figurile II.6 și II.8 arată performanța obținută începând de la momentul la care viteza plutonului este în jurul punctului de operare (i.e., toate vehiculele au atins viteza de $20m/s$). În figurile II.7 și II.9 sunt prezentate doar rezultatele pentru schimbarea de referința de la $20m/s$ la $30m/s$, pentru ieșiri și intrări. În toate figurile, în prima fereastră sunt prezentate vitezele plutonului cu referința corespunzătoare, în a doua fereastră sunt date mărimile de intrare (i.e., forțele de tracțiune), iar în a treia fereastră sunt afișate doar distanțele vehiculelor urmăritoare cu referința inpusă. Timpul discret este măsurat în secunde și este afișat pe axa orizontală.

Rezultatele arată că valoarea ponderilor intrării folosite în problema de optimizare influențează valoarea marimilor de intrare. Dacă penalizarea intrării este mare (i.e., $R_{u_i} = 0.001$), răspunsul în problema de control a distanței are oscilații mari în jurul referinței in-puse, obținut pentru o intrare mai mic, după cum este văzut în figura II.7. Rezultatul se poate compara cu răspunsul obținut pentru ponderea $R_{u_i} = 0.0005$ și prezentat în figura II.9. Explicația este dată de balanțarea caracteristică problemelor de optimizare între problema urmăririi referinței cu o eroare cât mai mică și un efort de control rezonabil. Este demn de remarcat faptul că în ambele scenarii de test, eroarea de distanța între vehicule scade spre finalul plutonului (i.e. datorită restricției terminale), ceea ce poate fi văzut ca o dovadă empirică de stabilitate a plutonului.

Performanța de urmărire a referinței pentru fiecare vehicul a fost evaluată folosind indicele eroare medie pătratică pentru viteză (MSE_{v_i}), $i = \overline{1, N}$:

$$MSE_{v_i} = \frac{1}{T_{end} - T_{start} + 1} \sum_{k=T_{start}}^{T_{end}} (ref_i(k) - v_i(k))^2 \quad (II.48)$$

unde $ref_i(k)$ și $v_i(k)$ notează referința inpusă și viteza măsurată pentru vehiculul i , la fiecare moment de eșantionare k . T_{start} marchează începutul intervalului de evaluare, iar

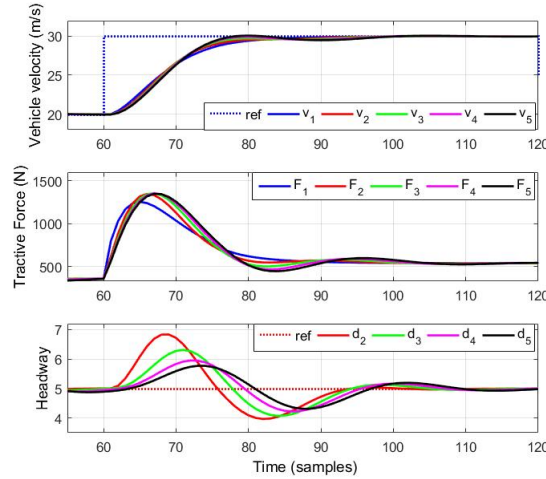


Figura II.7: Detaliu la schimbarea referinței pentru rezultatele în simulare obținute pentru intrarea ponderată cu valoarea $R_{u_i} = 0.001$

T_{end} corespunde cu finalul evaluării. Indicele de performanță a fost evaluat din momentul în care vehiculele au atins punctul de echilibru (i.e. de la timpul $T_{\text{start}} = 55$ s până la momentul $T_{\text{end}} = 180$ s).

Pentru vehiculul lider a fost predefinit un profil de referință a vitezei, în timp ce pentru fiecare vehicul următor, referința de viteză este valoarea vitezei vehiculului predecesor ($ref_i(k) = v_{i-1}(k)$).

Pentru o analiză completă, a fost calculată și media aritmetică a intrării $MEAN_{F_i}$, $i = \overline{1, N}$:

$$MEAN_{F_i} = \frac{1}{T_{\text{end}} - T_{\text{start}} + 1} \sum_{k=T_{\text{start}}}^{T_{\text{end}}} F_i(k) \quad (\text{II.49})$$

unde F_i este forța de tracțiune sau mărimea de control pentru vehiculul i , $i = \overline{1, N}$.

Ambii indici de performanță sunt dați în Tabelul 3, unde indicele numeric notează poziția vehiculului în pluton, pornind de la liderul plasat pe prima poziție.

Din indicele MSE , se poate observa faptul că pentru o valoare mai mică pentru ponderea intrării, se obține o urmărire a referinței mai bună în problema controlului vitezei de croazieră, cu prețul plătit într-o creștere ușoară a comenzii, vizibil în indicele $MEAN$. Aceste rezultate nu sunt surprinzătoare, deoarece problema de optimizare are doi termeni, iar ponderea intrării accentuează 'importanța' valorii mărimii de comandă, în timpul minimizării erorii de urmărire.

II.2.2.5 Concluzii

În acest capitol, a fost propus un algoritm DMPC pentru controlul în cadrul unui pluton atât a vitezei de croazieră cât și a distanței între vehicule. Fiecare agent responsabil pentru un vehicul următor, primește de la vehiculul predecesor comanda optimă fără întârzieri (i.e., într-o perioadă de eșantionare, informația relevantă este primită de la predecesor, este folosită în problema locală de optimizare și soluția optimă locală este transmisă către

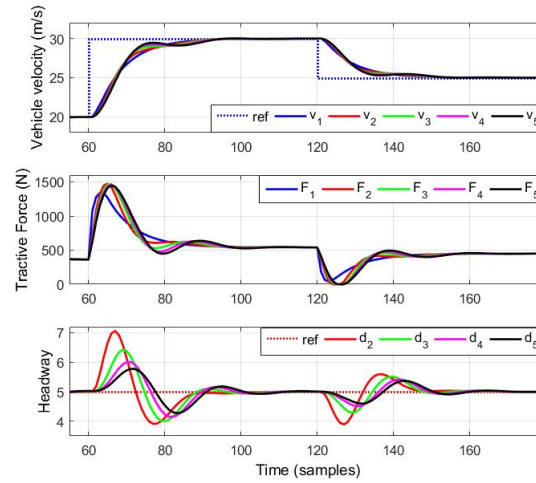


Figura II.8: Rezultate în simulare obținute pentru intrarea ponderată cu valoarea $R_{u_i} = 0.0005$.

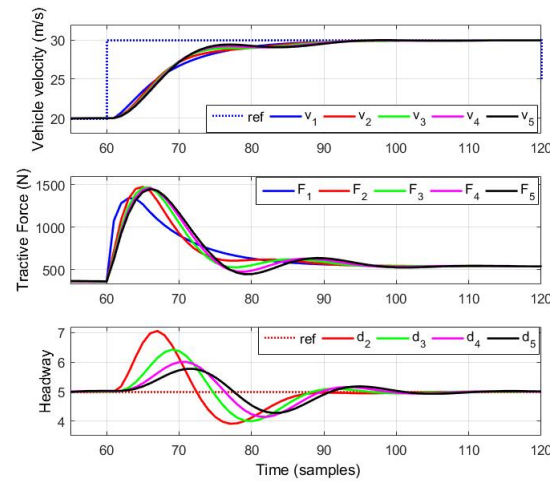


Figura II.9: Detaliu la schimbarea referinței pentru rezultatele în simulare obținute pentru intrarea ponderată cu valoarea $R_{u_i} = 0.0005$.

Table 2: Indicii de performanță MSE și $MEAN$ obținuți în cele două scenarii testate.

| index | $R_{u_i} = 0.001$ | $R_{u_i} = 0.0005$ |
|--------------|-------------------|--------------------|
| MSE_{v_1} | 6.3764 | 5.8453 |
| MSE_{v_2} | 0.0196 | 0.0238 |
| MSE_{v_3} | 0.0097 | 0.0118 |
| MSE_{v_4} | 0.0057 | 0.0065 |
| MSE_{v_5} | 0.0041 | 0.0042 |
| $MEAN_{F_1}$ | 522.5024 | 522.6604 |
| $MEAN_{F_2}$ | 522.5437 | 522.6636 |
| $MEAN_{F_3}$ | 522.5553 | 522.6888 |
| $MEAN_{F_4}$ | 522.4334 | 522.7703 |
| $MEAN_{F_5}$ | 522.1457 | 522.9019 |

sucesor). Au fost testate doua scenarii de simulare, cu ponderări ale comenzii diferite, și rezultatele au fost comparate folosind indicii *MSE* și *MEAN*. Rezultatele prezentate au arătat că trebuie făcut un compromis între performanța obținută în problema de urmărire a referinței și valoarea mărimii de comandă.

II.2.3 Algoritm de control predictiv distribuit coalițional cu formulare min-max

În această secțiune se propune o noua strategie de control robustă, formulată sub forma unui algoritm predictiv distribuit coalițional care asigură garanția fezabilității soluției optime formulate pentru un model cu incertitudini aditive mărginite. Această formulare, potrivită pentru sub-sisteme cuplate prin intrări, consideră variabilele de cuplare ca perturbații și asigură un consens robust cu un minim de informații transmise. Rezultatele obținute în simulare arată faptul că metoda coalițională are comportament similar cu algoritmul complet centralizat și are performanțe îmbunătățite față de algoritmul descentralizat și cel predictiv distribuit iterativ robust min-max. Rezultatele prezentate în această secțiune vor fi publicate în două conferințe [151, 152]

II.2.3.1 Introducere

Controlul predictiv distribuit MPC (eng. Model Predictive Control) este o strategie de control optimă capabilă să lucreze cu restricții și incertitudini explicit în formularea problemei de optimizare [101]. Cea mai conservativă abordare din literatura MPC, capabilă să facă față prezenței incertitudinilor este strategia min-max, care minimizează secvența de comandă, maximizând perturbațiile. În acest mod se optimizează scenariu în cel mai rău caz, în care incertitudinile sunt modelate cu limite deterministice [153, 154]. Conform [155], schemele min-max MPC pot fi clasificate în *i) control în buclă deschisă*, unde minimizând o funcție cost în cel mai rău caz, de obține o singură lege de reglare pentru toate perturbațiile admisibile [156] sau *ii) control în buclă închisă*, în care o funcție cost în cel mai rău caz este minimizată de-a lungul unei secvențe de legi de reglare [157].

Această metodologie a fost dezvoltată la finele anilor 60's [158] și este încă prezentă în literatură. Lucrări recente includ [159], în care se combină estimarea stărilor și control într-o singură problema min-max. În [160] este propus un regulator min-max MPC în buclă deschisă cu incertitudini aditive mărginite și o funcție de cost pătratică, în timp ce în [161] este prezentată o strategie min-max MPC în buclă închisă cu o funcție de cost pătratică. Ideea este reformularea problemei originale într-o problemă pătratică multi-parametrică, și obținerea unei soluții min-max explicite. În [162] este oferită o soluție în buclă închisă aproximativă, unde secvența de intrare viitoare este parametrizată în termeni ai valorilor viitoare ale perturbației, în locul parametrizării în termeni de stări viitoare descrisă în [163]. O formulare min-max MPC cu stabilitate intrare-ieșire bazată pe funcții Lyapunov, pentru sisteme cu incertitudini mărginite este propusă în [164], iar în [165] este descris un algoritm descentralizat, min-max MPC în buclă deschisă cu stabilitate intrare-la stare, pentru sisteme neliniare. Acesta din urmă, propune formularea interacțiunilor dintre sub-sisteme ca perturbații, și demonstrează stabilitatea prin proprietăți de robustețe.

Toate caracteristicile disponibile în MPC, pot fi extinse și la formulări descentralizate [166], spre exemplu controlul predictiv distribuit. În [167] este propus un algoritm min-

max DMPC, în care fiecare agent primește informația de cuplare pe stare ca și o perturbație în modelul local. Se propune o procedură iterativă, în care restricțiile pe stări și limitele incertitudinilor sunt actualizate la fiecare iterație. În acest algoritm, se folosește setul admisibil predictat ca noua valoare pentru limită.

Într-o strategie DMPC se partiționează sistemul într-un număr fix de sub-sisteme, bazat pe legăturile de cuplare inerente și pe structura rețelei de comunicare pre-aranjată dintre agenți. Această metodologie, deși funcționează, poate limita performanțele obținute în structura de control. Pentru a evita aceste neajunsuri, controlul MPC coalitional (C-MPC) oferă o nouă metodologie care asigură o flexibilitate ridicată a controlului [168].

Raționala de a permite agenților să formeze coaliții atunci când circumstanțele o cer a apărut din jocurile cooperative din domeniul teoriei jocurilor [169]. Astfel, atunci când condițiile de cuplare dintre sub-sisteme se schimbă, agenții implicați se pot alatura într-un grup cooperativ (numit coaliție), rezolvând o problemă de optimizare comună. În funcție de mecanismul folosit în formarea coalițiilor, topologiile rezultante se clasifică în [168]: *i) arhitecturi sus-jos*, dacă se folosește un nivel supervisor pentru a decide formarea coaliției [170, 171] sau *ii) arhitectura jos-sus*, dacă decizia de a forma o coaliție este luată la nivelul agenților locali, fără a folosi informație centralizată [172].

În această secțiune se propune o nouă strategie C-MPC cu actualizări iterative a limitelor incertitudinilor, bazată pe o formulare min-max DMPC în buclă deschisă. Cu toate că această abordare este conservativă, se dorește îmbunătățirea performanței față de formularea descentralizată, cu un minim de informație schimbat între agenți. Totuși, atunci când limitele incertitudinilor transmise între agenți depășesc un prag ales, atunci se încurajează formarea unei coaliții între agenți.

Algoritmul propus este formulat pentru o clasă de sub-sisteme cuplate prin intrări. Fiecare agent considera informația primită de la vecini ca o incertitudine marginată, care trebuie să fie sub un prag dat. Dacă perturbația este prea mare pentru a fi manipulată în problema de optimizare locală se formează o coaliție între vecini. Pe de altă parte, dacă incertitudinile primite sunt sub limită, atunci se pornește o procedură iterativă pentru a maximiza performanțele și a reduce limitele restricțiilor perturbațiilor. După un număr de iterații permis, primul element din secvența de comandă optimă este aplicat sub-sistemului, conform principiului orizontului alunecător. Această metodologie este relaționată cu alte rezultate din literatura de specialitate, e.g., [173, 174]. Cu toate acestea, algoritmul propus este coalitional și cantitatea de informație schimbată între agenți este minimizată prin definirea perturbațiilor ca având limite fixe.

II.2.3.2 Algoritmul MIN-MAX descentralizat

Fie un sistem care poate fi partiționat în N sub-sisteme cuplate dinamic prin intrări. Fiecare sub-sistem i , $\forall i \in \mathcal{N}$, unde \mathcal{N} notează setul $\{1, \dots, N\} \subseteq \mathbb{N}$, are un vector de stări $x_i \in R^{n_i}$, un vector de intrări $u_i \in R^{m_i}$, un vector de incertitudini $w_i \in R^{p_i}$ și un vector de ieșiri $y_i \in R^{q_i}$, și este descris de următoarele ecuații:

$$\begin{aligned} x_i(k+1) &= A_{ii}x_i(k) + B_{ii}u_i(k) + w_i(k) \\ w_i(k) &= \sum_{j \in \mathcal{N}_i \setminus \{i\}} B_{ij}u_j(k) \\ y_i(k) &= C_i x_i(k), \forall i \in \mathcal{N} \end{aligned} \tag{II.50}$$

unde A_{ii} , B_{ii} , B_{ij} and C_i sunt matrici de dimensiuni adecvate. Setul $\mathcal{N}_i = \{j \in \mathcal{N} : B_{ij} \neq 0\}$ conține toți vecinii sub-sistemului i , incluzând i . Astfel, agentul i , care este un regulator local responsabil cu sub-sistemul i , primește informații de la agentul j , dacă $j \in \mathcal{N}_i$, și transmite informații către agentul j if $i \in \mathcal{N}_j$.

S-au considerat următoarele restricții pe ieșiri, intrări și incertitudini:

$$y_i \in \mathcal{Y}_i, u_i \in \mathcal{U}_i, w_i \in \mathcal{W}_i, \forall i \in \mathcal{N} \quad (\text{II.51})$$

unde seturile \mathcal{Y}_i , \mathcal{U}_i și \mathcal{W}_i sunt definite de un set de inegalități liniare. E Rețineți că $\mathcal{W}_i = \bigoplus_{j \in \mathcal{N}_i \setminus \{i\}} B_{ij} \mathcal{U}_j$, unde \bigoplus denotă suma Minkowski.

Pentru fiecare agent $i \in \mathcal{N}$, se definește o problemă min-max locală, de optimizare cu restricții:

$$J_i(x_i^0) = \min_{u_i^0, \dots, u_i^{N_p-1}} \max_{w_i^0, \dots, w_i^{N_p-1}} \sum_{l=0}^{N_p-1} \|r_i^{l+1} - y_i^{l+1}\|_1 + \|R_{u_i} u_i^l\|_1 \quad (\text{II.52})$$

subject to (II.50) and (II.51) for $l \in [0, \dots, N_p]$

unde $y_i^l = y_i(k+l|k)$ denotă valorile viitoare ale ieșirii sub-sistemului i , predictate în cadrul orizontului de timp l , postulate la momentul de timp k , cu N_p fiind orizontul de predicție; secvența se obține pornind de la starea inițială $x_i^0 = x_i(k)$ măsurată la momentul de timp k , și aplicând modelului (II.50) secvența de intrare $\{u_i^0, \dots, u_i^{N_p-1}\}$ și setul conținând secvențele incertitudinilor $\mathcal{W}_i = \bigoplus_{j \in \mathcal{N}_i \setminus \{i\}} B_{ij} \mathcal{U}_j$; r_i^l este valoarea de referință impusă ieșirii;

$\|u_i^l\|_1$ denotă norma-1 în R^{m_i} și $R_{u_i} \in R^{m_i \times m_i}$ este o matrice nesingulară.

Secvența de intrare optimală U_i^* se obține rezolvând problema de optimizare min-max (II.52) cu o formulare în buclă deschisă, ceea ce înseamnă că intrarea minimizează cel mai rău caz al perturbațiilor.

II.2.3.3 Algoritmul MIN-MAX distribuit

În secțiunea II.2.3.2, a fost prezentată problema de optimizare descentralizată (II.52), în care fiecare agent i consideră intrările de cuplare ca fiind incertitudini. Soluția optimală locală este calculată fără nici un schimb de informații între vecini, anticipând cea mai gravă perturbație din partea acestora.

Fie restricțiile pe intrări (II.51) definite ca:

$$u_i^{\min} \leq u_i^l \leq u_i^{\max}, l = 0, \dots, N_p - 1 \quad (\text{II.53})$$

Asemenea,

$$w_i^{\min} \leq w_i^l \leq w_i^{\max}, l = 0, \dots, N_p - 1 \quad (\text{II.54})$$

cu,

$$\begin{aligned} w_i^{\min} &= \arg \min_{U_j} \bigoplus_{j \in \mathcal{N}_i \setminus \{i\}} B_{ij} \mathcal{U}_j \\ w_i^{\max} &= \arg \max_{U_j} \bigoplus_{j \in \mathcal{N}_i \setminus \{i\}} B_{ij} \mathcal{U}_j \end{aligned} \quad (\text{II.55})$$

unde $\mathcal{U}_j = [u_j^{\min}; u_j^{\max}]$ este definit ca o cutie (un paralelogram) cu limitele minime și maxime date de u_j^{\min} și respectiv u_j^{\max} . Se presupune ca aceste limite sunt nenegative.

Algoritmul min-max DMPC propus pornește de la formularea descentralizată și introduce două noi variabile de optimizare spre a fi minimizate. Acestea sunt limitele minime și maxime auto-impuse pentru variabilele de control locale, denotate cu \hat{u}_i^{\max} și \hat{u}_i^{\min} . Întrucât, sub-sistemele sunt cuplate prin intrări, valorile acestor variabile sunt schimbate între vecini la fiecare iterație. În această manieră, fiecare agent se angajează ca traiectoria de intrare optimală actuală, care va acționa ca o perturbație transmisă către vecini, va fi între noile limite auto-impuse. Având această nouă informație disponibilă, regulatoarele reevaluează problema de optimizare locală, folosind valorile limitelor auto-impuse primite de la vecini, ca și limite de restricții pentru incertitudini în problema de optimizare, respectiv limitele auto-impuse calculate local fiind folosite ca și limite pentru restricțiile formulate pentru variabilele de control locale.

Astfel, problema de optimizare min-max DMPC ce va fi rezolvată de fiecare agent i , $\forall i \in \mathcal{N}$ este:

$$\begin{aligned} J_i(x_i^0) &= \min_{\substack{u_i^0, \dots, u_i^{N_p-1} \\ \hat{u}_i^{\min}, \hat{u}_i^{\max}}} \max_{w_i^0, \dots, w_i^{N_p-1}} \|R_{w_i} \hat{u}_i^{\min}\|_1 + \|R_{w_i} \hat{u}_i^{\max}\|_1 \\ &+ \sum_{l=0}^{N_p-1} \|r_i^{l+1} - y_i^{l+1}\|_1 + \|R_{u_i} u_i^l\|_1 \end{aligned} \quad (\text{II.56})$$

s. t. (II.50)

$$\begin{aligned} y_i^l &\in \mathcal{Y}_i \\ u_i^{\min} &\leq \hat{u}_i^{\min} \leq u_i^l \leq \hat{u}_i^{\max} \leq u_i^{\max} \\ \hat{u}_i^{\min}, \hat{u}_i^{\max} &\geq 0 \\ w_i^{\min} &\leq w_i^l \leq w_i^{\max} \end{aligned}$$

Aici, $R_{w_i} \in R^{m_i \times m_i}$ este matricea de ponderare pentru limitele auto-impuse ale variabilelor de control.

Algoritmul min-max DMPC propus poate fi sumarizat ca:

Algoritm 1

- La fiecare perioadă de eșantionare k , fiecare agent i , $\forall i \in \mathcal{N}$ primește de la senzori măsurătorile variabilelor de stare.
- La iterația $p = 0$:
 - Pas 1: Calculează valorile pentru $U_i^{*,0}$, $\hat{u}_i^{\min,0}$, $\hat{u}_i^{\max,0}$ prin minimizarea problemei de optimizare (II.56) folosind valorile implicite ale limitelor restricțiilor (u_i^{\min} , u_i^{\max} , w_i^{\min} , w_i^{\max}).

- Pas 2: Transmite valorile $\hat{u}_i^{\min,0}, \hat{u}_i^{\max,0}$ către vecinii j , dacă $i \in \mathcal{N}_j$ și primește de la vecinii $j \in \mathcal{N}_i \setminus \{i\}$ valorile corespunzătoare pentru $\hat{u}_j^{\min,0}, \hat{u}_j^{\max,0}$

- Execută

- Pas 1: Calculează valorile pentru $U_i^{*,p}, \hat{u}_i^{\min,p}, \hat{u}_i^{\max,p}$ prin minimizarea problemei de optimizare (II.56) folosind noile limite pentru restricții:

$$\begin{aligned} u_i^{\min,p} &= \hat{u}_i^{\min,p-1}, \quad u_i^{\max,p} = \hat{u}_i^{\max,p-1} \\ w_i^{\min,p} &= \arg \min_{U_j^{p-1}} \bigoplus_{j \in \mathcal{N}_i \setminus \{i\}} B_{ij} \mathcal{U}_j^{p-1} \\ w_i^{\max,p} &= \arg \max_{U_j^{p-1}} \bigoplus_{j \in \mathcal{N}_i \setminus \{i\}} B_{ij} \mathcal{U}_j^{p-1} \\ U_j^{p-1} &= [\hat{u}_j^{\min,p-1}; \hat{u}_j^{\max,p-1}], \quad j \in \mathcal{N}_i \setminus \{i\}. \end{aligned}$$

- Pas 2: Transmite valorile $\hat{u}_i^{\min,p}, \hat{u}_i^{\max,p}$ către vecini și primește valorile corespunzătoare $\hat{u}_j^{\min}, \hat{u}_j^{\max}$ pentru $j \in \mathcal{N}_i \setminus \{i\}$
- Pas 3: Dacă toate problemele sunt fezabile, treci la iterația $p = p + 1$ și repetă algoritmul începând cu Pasul 1 până când se atinge convergența soluției sau un număr maxim de iterații este realizat. Altfel folosește soluția de la iterația precedentă.
- Pas 4: Aplică sub-sistemului primul element al secvenței de comandă optimală $U_i^{*,p}$ și așteaptă până la momentul de eșantionare $k + 1$.

Rețineți că există un minim de informație schimbat între agenți, și anume cutia definită de limitele auto-impuse pentru restricții. În cele din urmă, putem presupune că pentru prima iterație ($p = 0$), toate problemele de optimizare sunt fezabile. Rețineți că aceasta corespunde cu fezabilitatea problemelor min-max MPC descentralizate.

II.2.3.4 Algoritmul MIN-MAX DMPC coalițional

Folosind cadrul algoritmului min-max DMPC prezentat în secțiunea II.2.3.3, în cele ce urmează se va introduce mecanismul coalițional. Ideea este ca fiecare agent să formeze o coaliție cu vecinii săi atunci când gradul de incertitudine primit de la vecini compromite performanța problemei de optimizare locale. Se folosește politica 'crează un prieten din cel mai mare dușman', sau vorba românească 'fă-te frate cu dracu până treci puntea', adică se propune o cooperare cu vecinul care crează perturbația cea mai mare.

În interiorul unei coaliții, problema de optimizare cooperativă se rezolvă având acces complet la informația schimbată între vecini. Fiecare agent $i \in \mathcal{N}$, poate forma o coaliție cu ceilalți agenți $j \in \mathcal{N}_i$ din vecinătatea sa. Când aceasta se întâmplă, setul de agenți $\mathcal{C} \subseteq \mathcal{N}_i$ pot forma o coaliție cu modelul:

$$\begin{aligned} x_{\mathcal{C}}(k+1) &= A_{\mathcal{C}\mathcal{C}}x_{\mathcal{C}}(k) + B_{\mathcal{C}\mathcal{C}}u_{\mathcal{C}}(k) + w_{\mathcal{C}}(k) \\ w_{\mathcal{C}}(k) &= \sum_{j \in \mathcal{N}_{\mathcal{C}} \setminus \{i\}} B_{\mathcal{C}j}u_j(k) \\ y_{\mathcal{C}}(k) &= C_{\mathcal{C}}x_{\mathcal{C}}(k) \end{aligned} \tag{II.57}$$

unde x_c , u_c , w_c și y_c denotă vectorii de stare, intrare, incertitudine și ieșire a coaliției, calculați prin agregarea vectorilor locali corespunzători (e.g., $x_c = [x_i]_{i \in \mathcal{C}}$). De asemenea, matricile A_{cc} , B_{cc} , B_{cj} și C_c sunt calculate conform agregării.

Seturile restricțiilor corespunzătoare coaliției sunt definite ca:

$$\begin{aligned} y_c \in \mathcal{Y}_c &= \prod_{i \in \mathcal{C}} \mathcal{Y}_i, \quad u_c \in \mathcal{U}_c = \prod_{i \in \mathcal{C}} \mathcal{U}_i, \\ w_c \in \mathcal{W}_c &= \prod_{i \in \mathcal{C}} \mathcal{W}_i, \quad \forall i \in \mathcal{N} \end{aligned} \quad (\text{II.58})$$

și problema de optimizare min-max rezolvată de către coaliție este similară cu (II.56):

$$\begin{aligned} J_c(x_c^0) &= \min_{\substack{u_c^0, \dots, u_c^{N-1} \\ \hat{u}_c^{\min}, \hat{u}_c^{\max}}} \max_{w_c^0, \dots, w_c^{N-1}} \|R_{w_c} \hat{u}_c^{\min}\|_1 + \|R_{w_c} \hat{u}_c^{\max}\|_1 \\ &\quad + \sum_{l=0}^{N_p-1} \|r_c^{l+1} - y_c^{l+1}\|_1 + \|R_{u_c} u_c^l\|_1 \end{aligned} \quad (\text{II.59})$$

s. t. (II.58)

$$\begin{aligned} y_c^l &\in \mathcal{Y}_c \\ u_c^{\min} &\leq \hat{u}_c^{\min} \leq u_c^l \leq \hat{u}_c^{\max} \leq u_c^{\max} \\ \hat{u}_c^{\min}, \hat{u}_c^{\max} &\geq 0 \\ w_c^{\min} &\leq w_c^l \leq w_c^{\max} \end{aligned}$$

Aici R_{u_c} și R_{w_c} sunt matrici bloc diagonale care conțin toate ponderile locale, r_c^l este vectorul agregat cu valorile impuse pentru referințe și x_c^0 este vectorul agregat cu stările inițiale.

Algoritmul min-max DMPC coalițional poate fi sumarizat ca:

Algoritm 2

- La fiecare perioadă de eșantionare k , fiecare agent i , $\forall i \in \mathcal{N}$ primește de la senzori măsurătorile variabilelor de stare.
- Pas 1: Calculează valorile inițiale pentru $U_i^{*,0}$, $\hat{u}_i^{\min,0}$, $\hat{u}_i^{\max,0}$ prin rezolvarea (II.56) folosind valorile implicite ale limitelor restricțiilor (u_i^{\min} , u_i^{\max} , w_i^{\min} , w_i^{\max}).
- Pas 2: Transmite valorile $\hat{u}_i^{\min,0}$, $\hat{u}_i^{\max,0}$ către vecinii j , dacă $i \in \mathcal{N}_j$ și primește de la vecinii $j \in \mathcal{N}_i \setminus \{i\}$ valorile corespunzătoare pentru $\hat{u}_j^{\min,0}$, $\hat{u}_j^{\max,0}$.
- Pas 3: Calculează cutia pentru perturbații, folosind valorile minime și maxime pentru fiecare perturbație, primite de la vecini:

$$\begin{aligned} w_i^{\min,0} &= \arg \min_{U_j^0} \bigoplus_{j \in \mathcal{N}_i \setminus \{i\}} B_{ij} \mathcal{U}_j^0 \\ w_i^{\max,0} &= \arg \max_{U_j^0} \bigoplus_{j \in \mathcal{N}_i \setminus \{i\}} B_{ij} \mathcal{U}_j^0 \\ U_j^0 &= [u_j^{\min,0}; u_j^{\max,0}], \quad j \in \mathcal{N}_i \setminus \{i\} \end{aligned}$$

- Pas 4: Verifică dacă nivelul de incertitudini primit de la vecini (i.e., $w_i^{\min,0}$, $w_i^{\max,0}$) depășește un prag γ ,
while $\max(|w_i^{\min,0}|, |w_i^{\max,0}|) > \gamma$
 Aadaugă la lista de coaliție vecinul care generează valoarea maximă de incertitudine, ceea ce implică faptul că se elimină contribuția sa din $w_i^{\min,0}$ and $w_i^{\max,0}$.
end
- Pas 5: Creează coalițiile rezultante, care sunt formate satisfăcând listele de coaliții generate la Pasul 4.
- Pas 6: Folosește Algoritmul 1 la nivel coalițional. Rețineți că aceasta implică rezolvarea pentru fiecare coaliție a problemei (II.59) în locul problemei (II.56). De asemenea, rețineți că coalițiile pot fi formate doar cu un singur agent.

Rețineți că în Pasul 6, când Algoritmul 1 este utilizat, problema de control coalițională este fezabilă, deoarece $\mathcal{W}_C \subseteq \prod_{i \in C} \mathcal{W}_i$ și $\mathcal{U}_C = \prod_{i \in C} \mathcal{U}_i$. Astfel, încă o dată algoritmul este fezabil dacă problema descentralizată min-max MPC este fezabilă. Este demn de menționat faptul că, dacă coaliția completă se formează ($C = \mathcal{N}$), atunci problema de optimizare coalițională robustă min-max (II.59) devine o problemă clasică MPC centralizată, deoarece incertitudinile dintre agenți dispar. În cele din urmă, ajustând pragul γ , se poate stabili gradul de cooperare dintre agenții vecini.

II.2.3.5 Rezultate

În cele ce urmează, se vor prezenta rezultatele obținute în simulare pentru algoritmul the min-max DMPC din secțiunea II.2.3.3 și algoritmul coalițional dat în secțiunea II.2.3.4. Pentru simplitate, a fost tratat doar cazul unui sistem compus din două sub-sisteme ($N=2$).

Fie exemplul academic conținând două dublu integratoare cuplate prin intrări luat din [175]. Primul sub-sistem este descris de următoarele matrici:

$$A_{11} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \quad B_{11} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$B_{12} = \begin{bmatrix} 0 \\ 0.4 \end{bmatrix} \quad C_1 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}^T$$

și al doilea sub-sistem este dat de:

$$A_{22} = \begin{bmatrix} 1 & 1 \\ 0 & 1 \end{bmatrix} \quad B_{21} = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

$$B_{22} = \begin{bmatrix} 0 \\ 1 \end{bmatrix} \quad C_2 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}^T$$

unde indexul superior T denotă operatorul matrice transpusă.

Ambele sub-sisteme au următoarele limite de restricții:

$$\begin{cases} u_i^{\min} = -1 \\ u_i^{\max} = 1 \end{cases} \quad \begin{cases} w_i^{\min} = -1 \\ w_i^{\max} = 1 \end{cases} \quad \begin{cases} y_i^{\min} = -10 \\ y_i^{\max} = 10 \end{cases} \quad (\text{II.60})$$

Table 3: ST (timp de raspuns), OV (suprareglare) și J^{cum} (cost cumulativ) pentru algoritmi min-max MPC descentralizat, min-max DMPC, DMPC coalitional și MPC centralizat

| Algoritm de control | ST | OV | J^{cum} |
|------------------------------|------|------|------------------|
| MPC descentralizat (Y_1) | 17 | 55 | 20.19 |
| min-max DMPC (Y_1) | 11 | 25 | 17.81 |
| DMPC coalitional (Y_1) | 6 | 0 | 17.56 |
| MPC centralizat (Y_1) | 6 | 0 | 17.53 |
| MPC descentralizat (Y_2) | 17 | 143 | 17.83 |
| min-max DMPC (Y_2) | 11 | 67 | 10.90 |
| DMPC coalitional (Y_2) | 5 | 0 | 6.99 |
| MPC centralizat (Y_2) | 5 | 0 | 6.90 |

Table 4: Valorile J^{cum} (sumă cumulativă totală) și $Sim Time$ (timp de simulare (s)) obținute în scenariile testate

| $MSE(m^2)$ | J^{cum} | $Sim Time(s)$ |
|---------------------|------------------|---------------|
| MPC Centralizat | 24.43 | 8.81 |
| DMPC Coalitional | 24.55 | 88.99 |
| MPC Decentralizat | 38.03 | 27.45 |
| min-max DMPC 0 iter | 38.00 | 31.89 |
| min-max DMPC 1 iter | 31.66 | 66.73 |
| min-max DMPC 2 iter | 28.71 | 105.64 |
| min-max DMPC 3 iter | 28.66 | 150.34 |
| min-max DMPC 4 iter | 28.64 | 228.33 |

Simulările au fost efectuate cu un calculator cu frecvența procesorului de 2.20 GHz, având 4 GB memorie RAM, în Matlab 2014, folosind toolbox-ul YALMIP pentru optimizare robustă [176], cu orizontul de predicție $N_p = 4$. Ponderile intrărilor sunt $R_{u_i} = 0.5$ și $R_{w_i} = 0.01$. Valorile inițiale pentru stare au fost alese ca $x_1(0) = [5 \ 0]^T$ și respectiv $x_2(0) = [0 \ 0]^T$, cu referința impusă $r_i = 1$ pentru $i = 1, 2$.

În figura II.10, sunt prezentate rezultatele obținute în experimentul de urmărirea referinței pentru algoritmul min-max descentralizat, iar în figura II.11 sunt date rezultatele algoritmului distribuit min-max DMPC cu două iterații. Deoarece Algoritmul 1 introduce două noi variabile de optimizare (limitele auto-impuse \hat{u}_i^{\min} și \hat{u}_i^{\max}), intrarea optimală este restricționată între aceste două limite. În figura II.11, aceste limite sunt marcate cu linie neagră întreruptă și respectiv punctată.

În experimentul min-max descentralizat se poate remarca un fenomen interesant. Cu toate că intrarea u_2 are valori pozitive în primele 5 eșantioane, ieșirea y_2 are un trend inițial descrescător și apoi începe să crească spre valoarea impusă a referinței, atunci când intrarea de cuplare u_1 atinge valori pozitive. În acest caz, cuplajul dintre sub-sisteme determină un conflict între agenți. Acest fenomen este rezolvat în algoritmul DMPC, deoarece valorile maxime posibile sunt schimbate între agenți.

Algoritmul coalitional DMPC cu pragul $\gamma = 0.5$ este afișat în figura II.12, iar în Figura II.13 sunt date rezultatele pentru algoritmul MPC centralizat. Momentele de timp k când se activează coaliția sunt marcate cu cercuri albastre. Performanța algoritmilor min-max DMPC și DMPC coalitional este validată cu algoritmi min-max MPC descentralizat și

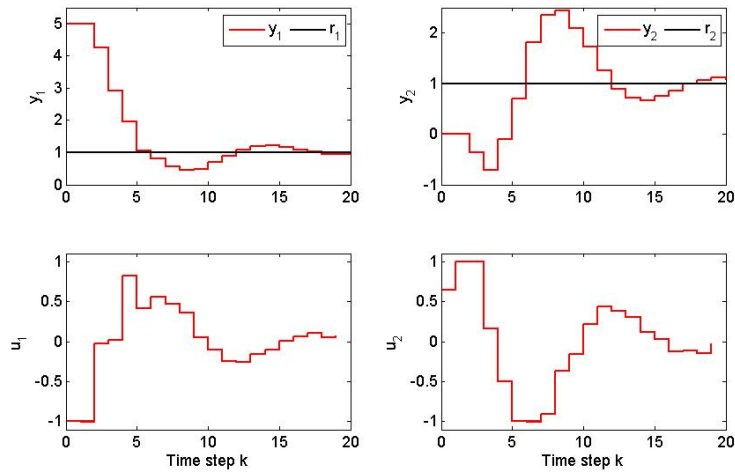


Figura II.10: Rezultate în simulare obținute cu algoritmul *min-max descentralizat* pentru o problemă de control cu urmărirea referinței

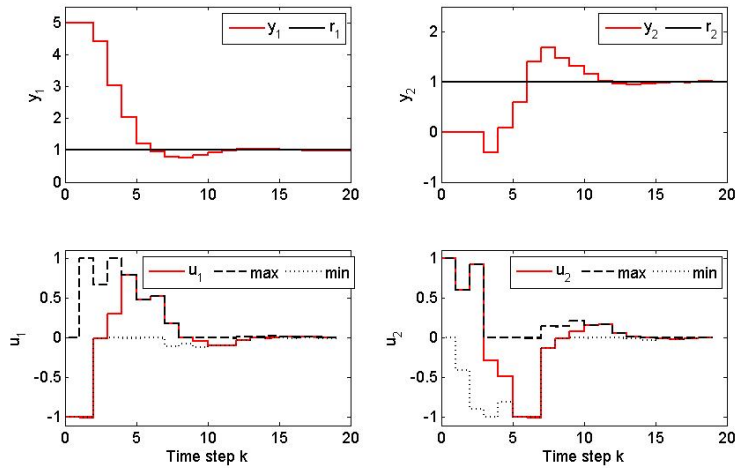


Figura II.11: Rezultate în simulare obținute cu algoritmul iterativ *min-max DMPC* pentru o problemă de control cu urmărirea referinței

MPC centralizat. Pentru evaluare au fost calculați următorii indici: timp de răspuns ST , suprareglare OV și cost cumulativ J_i^{cum} :

$$J_i^{\text{cum}} = \|\bar{Y}_i\|_1 + R_{u_i} * \|\bar{U}_i\|_1 \quad (\text{II.61})$$

unde $i = 1, 2$ este indicele sub-sistemului. \bar{Y}_i și \bar{U}_i sunt vectori cu ieșirea măsurată și respectiv intrarea pe durata întregii simulări. Performanța este dată în Tabelul 3.

Rezultatele arată că algoritmul iterativ min-max DMPC are performanțe mai bune decât algoritmul descentralizat: cu 35% îmbunătățire a timpului de creștere, 54% a suprareglării și 24% a costului cumulativ total. De asemenea, raportat la algoritmul MPC centralizat, algoritmul coalițional DMPC are o creștere în procente de 0% pentru timpul de creștere,

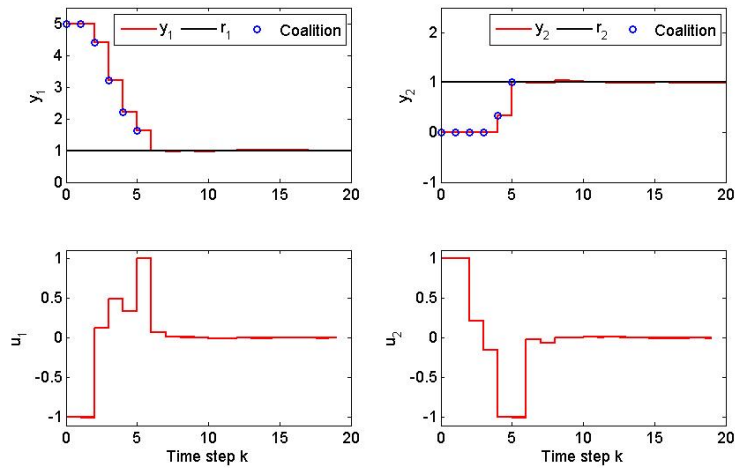


Figura II.12: Rezultate în simulare obținute cu algoritmul min-max coalițional propus *C-MPC* pentru o problemă de control cu urmărirea referinței

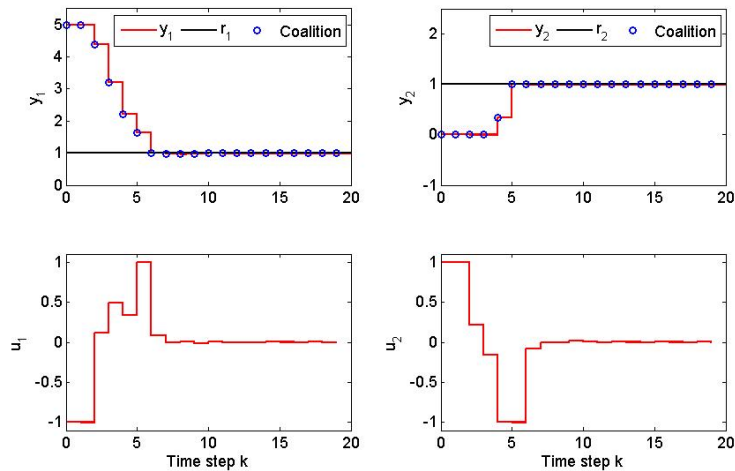


Figura II.13: Rezultate în simulare obținute cu algoritmul MPC *centralizat* pentru o problemă de control cu urmărirea referinței

0% pentru suprareglare și 0.49% pentru costul cumulativ total. Rețineți că în testul DMPC coalițional, atunci când coaliția se formează, se rezolvă o problemă de optimizare centralizată, iar în restul simulării se execută algoritmul iterativ DMPC, cu două iterații. În acest caz, Algoritmul 1 este terminat după două iterații.

Deoarece se propune un algoritm min-max distribuit iterativ, a fost evaluată și influența numărului de iterații asupra performanțelor obținute. Din punctul de vedere al costului cumulativ total, rezultatele sumarizate în Tabelul 4 arată faptul că indicele de performanță descrește cu numărul de iterații, iar timpul de simulare necesar întregului experiment crește. Aceste rezultate dovedesc că trebuie realizat un compromis între performanța dorită și efor-

tul de calcul, deoarece creșterea numărului de iterații la trei sau patru iterații, nu îmbunătățește costul și nu justifică creșterea timpului de simulare. Totuși, este demn de menționat faptul că toate strategiile de control prezentate au fost testate pe un singur calculator și că timpul de simulare al întregului experiment ar putea descrește dacă fiecare agent ar fi evaluat în paralel pe calculatoare diferite.

Așa cum era de așteptat, algoritmul MPC centralizat are cel mai bun cost cumulativ și timp de simulare, deoarece la fiecare iterație se rezolvă o singură problemă de optimizare, urmat îndeaproape de algoritmul coalițional DMPC propus, care îmbunătățește rezultatele obținute cu algoritmul iterativ DMPC cu 16% și 18% în termeni de cost cumulativ și respectiv timp de simulare.

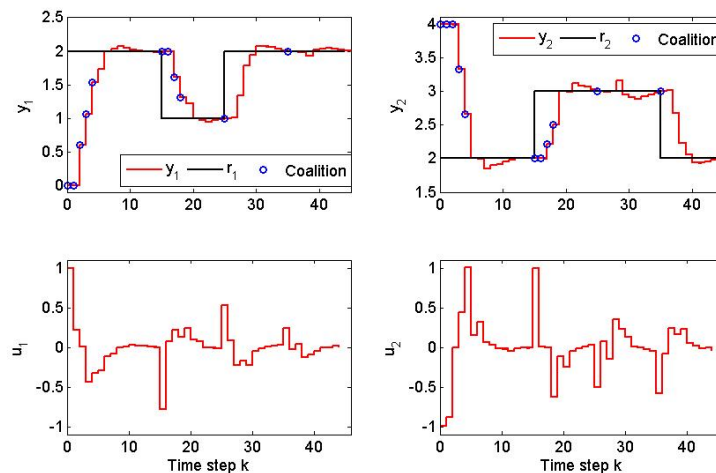


Figura II.14: Rezultate în simulare obținute cu algoritmul min-max coalițional propus *C-MPC* pentru o problemă de control cu mai multe schimbări de referință

În cele din urmă, în figura II.14 este prezentat rezultatul testului în simulare pentru un experiment cu mai multe schimbări de referință. S-a folosit valoarea de prag $\gamma = 0.3$. Rezultatele arată faptul că agenții tind să formeze coaliții consecutive atunci când au strategii de control în conflict, e.g., la începutul experimentului și la eșantionul de timp 15, deoarece schimbările de referință sunt simultane și în direcții opuse. De asemenea, la eșantioanele de timp 25 and 35 se formează coaliția atunci când referințele se schimbă.

II.2.3.6 Concluzii

În această secțiune, s-a introdus o strategie de control min-max MPC coalițional distribuit. Ideea principală este dată de calculul iterativ a unor noi limite pentru restricții, prin intermediul minimizării unei probleme de optimizare locale, și transmiterea acestei informații către agenții vecini.

Rezultatele obținute în simulare arată că algoritmul coalițional MPC propus are performanțe mai bune decât varianta iterativă a algoritmului distribuit, care la rândul său are per total rezultate mai bune decât algoritmul min-max descentralizat. Fezabilitatea schemei de control a fost generată pornind de la niște presupuneri inițiale ușor de îndeplinit.

Rețineți faptul că în cazul controlului unui pluton, variabilele de cuplare dintre vehicule pot fi considerate incertitudini la nivelul problemelor de optimizare locale, algoritmi propuși putând fi astfel direct generalizați pentru această aplicație.

II.2.4 Sisteme cyber fizice - proiectarea orientată a controlului cooperativ pentru plutoane de vehicule

În industria automotive de astăzi dezvoltarea produselor hardware și software a crescut așa de mult încât rețelele dintr-un singur vehicul pot include până la 100 de ECU-uri, senzori și elemente de execuție care sunt capabile să execute un număr mare de bucle de reglare și să măsoare o cantitate mare de date. Pe măsură ce ECU-urile devin mai puternice, acestea sunt expuse în mod continuu la efectele secundare ale tehnologiilor semiconductorilor sculate ca și erori în soft, variabilitatea de fabricație și îmbătrânirea [177]. Considerând tehnicile de proiectare adecvate pentru algoritmi de control, o parte dintre efectele îmbătrânirii pot fi reduse. În același timp, numărul mare de ECU-uri dintr-o rețea vehiculară permite creșterea automatizării vehiculelor și includerea lor în Sistemele de Transport Inteligent. Astfel, o serie de probleme critice care devin din ce în ce mai serioase în cazul sistemelor de transport, cum sunt accidentele rutiere, blocajele în trafic și poluarea [178] pot fi rezolvate.

Construirea mai multor drumuri și mai mari pentru a scăpa de problema ambuteiajelor nu este o metodă eficientă deoarece costurile sunt enorme și terenul disponibil este limitat. O soluție acceptată poate fi reprezentată de gruparea vehiculelor în plutoane. Principalele beneficii care sunt aduse de organizarea vehiculelor în plutoane sunt: creșterea capacității drumurilor, scăderea consumului de energie și a emisiilor, creșterea confortului și a siguranței șoferului și facilitarea dezvoltării aplicațiilor bazate pe comunicațiile cooperativ datorită poziției relativ fixe pentru vehiculele din același pluton [178].

Sistemele cu o interacțiune strânsă între unitățile computaționale (cyber) și sistemele fizice sunt cunoscute ca și sisteme cyber-fizice (CPS) [179]. Acestea implică o bună colaborare între strategia de control, sistemele încorporate, sistemele de timp real și ingineria software. Cu aplicarea asupra plutoanelor de vehicule astfel de sisteme complexe sunt denumite sisteme cyber-fizice vehiculare bazate pe plutoane (VCPS), în care toate vehiculele comunică prin intermediul unor rețele wireless și se deplasează în formația de tip pluton, cu o buclă de reglare închisă de tip feedback între procesul cyber și cel fizic [178].

În literatura de specialitate există o serie de lucrări care constau în cercetarea pe baza controlului predictiv a plutoanelor de vehicule. Autorii lucrării [143] tratează problema de control a unui convoi împărțind-o într-o problemă de nivel superior (nivelul plutonului) și alta de nivel inferior (nivelul vehiculului). Pentru aceasta, aceștia folosesc o strategie de control predictiv bazată pe model (MPC). La nivelul superior, este utilizat MPC pentru generarea traiectoriei centrului virtual al vehiculelor. La nivelul inferior, regulatorul MPC asignat are scopul de a calcula intrările vehiculului în vederea urmăririi traiectoriei generate. În [180] un algoritm de control predictiv distribuit (DMPC) este propus pentru controlul plutoanelor de vehicule eterogene cu topologii unidirecționale. Vehiculele dintr-un pluton, considerate ca și noduri într-un graf, sunt decuplate dinamic, dar având

limitări date de geometria spațială. Fiecare nod se bazează doar pe informația primită de la vecinii acestuia și utilizează această informație într-o buclă deschisă locală cu o funcție de cost proiectată să penalizeze erorile dintre traiectoriile predictate și cele presupuse. Lucrarea [181] studiază adaptarea inteligentă a vitezei plutoanelor pe autostradă combinând inteligența vehiculelor automate cu cea a infrastructurii rutiere. Intenția este de a determina cum poate găsi regulatorul ce ține de infrastructură valori optime pentru viteza plutoanelor utilizând MPC. În [182] este propusă proiectarea unui regulator predictiv generalizat (GPC) bazat pe un model CARIMA pentru a controla sistemul de direcție a unui singur vehicul pentru a asigura urmărirea unei anumite căi.

Principalul scop al acestei lucrări este proiectarea orientată pe sistemele cyber-fizice a unui sistem de control cooperativ pentru plutoane de vehicule. Acest tip de sistem are două planuri: planul fizic care descrie mobilitatea plutonului sub constrângerile traficului și planul cyber care descrie comportamentele rețelelor de vehicule formate din vehicule adiacente. Principalul obiectiv de control pentru un pluton de vehicule este de a urmări vehiculul precedent la o anumită distanță care depinde de viteza vehiculului. Plutonul este compus dintr-un lider al cărui viteză este controlată prin intermediul unui sistem cruise control (CC) implementat cu un regulator PID, și un număr finit de urmăritori cu sisteme adaptive de control a vitezei de croazieră (ACC) care utilizează regulatoare GPC cu tratarea efectului perturbațiilor măsurabile pe calea feed-forward pentru ajustarea distanței între vehicule. Sunt considerate comunicații între vehicule pentru a transmite viteza de la un vehicul la succesorul acestuia și de a dezvolta o structură de control cooperativ adaptivă pentru viteza de croazieră (CACC) pentru pluton utilizând abordări bazate pe sisteme CPS. Această viteză este considerată ca o perturbație măsurabilă și este rejectată folosind componenta feedforward a algoritmului predictiv de control. Performanțele plutonului au fost testate prin simulare considerând un pluton cu un lider și 15 urmăritori.

II.2.4.1 Modelul vehiculului

Modelul vehiculului a fost determinat pornind de la ecuația de mișcare longitudinală [138]:

$$m \frac{dv_v}{dt} = F_x - mg \sin \theta - fmg \cos \theta - 0.5\rho AC_d(v_v + v_w)^2, \quad (\text{II.62})$$

unde m este masa vehiculului, F_x este forța de tracțiune, g este accelerația gravitațională, θ este înclinarea drumului, f este coeficientul de rezistență la rulare, ρ este densitatea aerului, A este aria frontală a vehiculului, C_d este coeficientul de frecare, v_v este viteza vehiculului și v_w este viteza vântului.

Liniarizând ecuația (II.62), parametrii modelului sunt definiți după cum urmează:

$$\begin{aligned} v_v &= v_0 + v'; F_x = F_{x0} + u; \theta = \theta_0 + \theta', \\ w_v &= mg(f \sin \theta_0 - \cos \theta_0) \theta', \\ \tau_v &= (m / (\rho C_d A (v_0 + v_w))), \\ K_v &= (1 / (\rho C_d A (v_0 + v_w))). \end{aligned} \quad (\text{II.63})$$

Cunoscând parametrii K_v și τ_v din relațiile (II.63) este posibilă construirea unei funcții

de transfer pentru simularea mișcării longitudinale a vehiculului:

$$G_v(s) = \frac{K_v}{\tau_v s + 1}. \quad (\text{II.64})$$

Pentru sistemele CC utilizate în această lucrare se intenționează construirea unui model al vehiculului mai realist pornind de la funcția de transfer a elementului de execuție care ajustează unghiul de accelerație bazat pe un motor de curent continuu introdus în [149]. Astfel modelul elementului de execuție devine:

$$G_a(s) = \frac{K_a}{s(\tau_a s + 1)} \quad (\text{II.65})$$

unde intrarea este duty-cycle-ul motorului (în procente), iar ieșirea este forța de tracțiune.

Luând în considerare funcțiile de transfer (II.64) și (II.65) a fost construit modelul vehiculului pentru sistemul CC din Figura II.15, iar funcția de transfer a părții fixate are următoarea formă:

$$G_l(s) = G_a(s)G_v(s) = \frac{K_a K_v}{s(\tau_a s + 1)(\tau_v s + 1)}, \quad (\text{II.66})$$

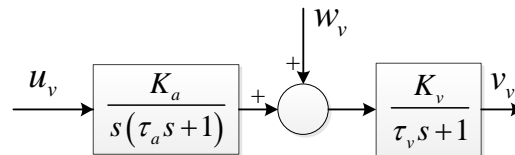


Figura II.15: Modelul vehiculului pentru sistemul CC

Pentru sistemele ACC, vehiculul urmăritor a fost modelat luând în considerare doar funcția de transfer (II.64) a mișcării longitudinale a vehiculului.

II.2.4.2 Controlul vehiculului pentru deplasarea în pluton

Problema de control a vehiculului constă în proiectarea unui sistem CACC orientat pe sisteme CPS pentru deplasarea în pluton. Așa cum este ilustrat în Figura II.15, structura de control a sistemului CACC este compusă dintr-un regulator CC pentru vehiculul lider și un regulator cu reacție negativă ACC cu compensator implicit al perturbației pe calea feed-forward pentru vehiculul urmăritor. Toate vehiculele sunt echipate cu sisteme de comunicații wireless (WCS) pentru a fi capabile să transmită/să recepționeze date spre/din-spre vehiculul de după/dinaintea acestora.

II.2.4.3 Metoda de control pentru liderul plutonului

Pentru controlul vitezei vehiculului lider este folosită o structură cu un regulator PID ca și cea din Figura II.17.

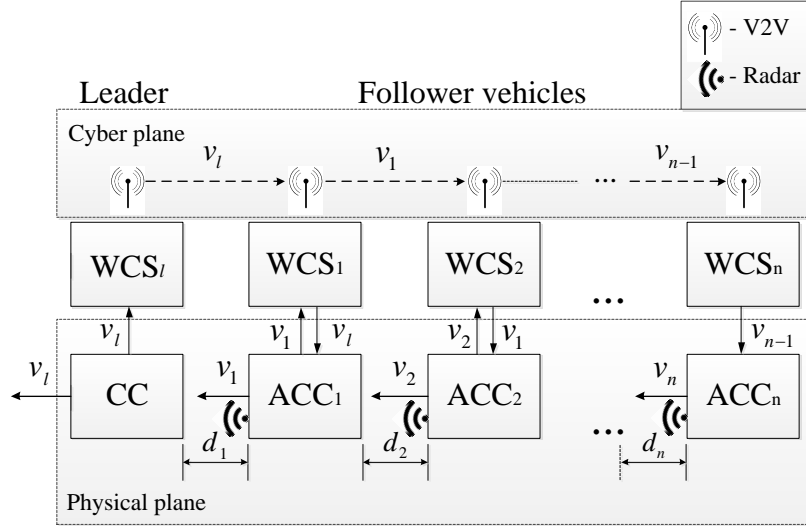


Figura II.16: Schema unui pluton cu sistemul CACC implementat

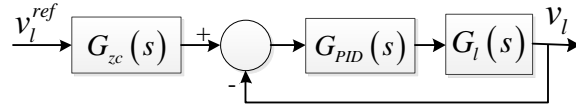


Figura II.17: Structura de control a liderului plutonului cu regulator PID

Regulatorul PID are următoarea formă paralelă [138]:

$$G_{PID}(s) = K_P + \frac{K_I}{s} + K_D s \quad (\text{II.67})$$

Ieșirea generată de regulator este duty cycle-ul motorului care controlează unghiul accelerației (în procente), care este transmisă către elementul de execuție. Funcția de transfer în buclă închisă considerând expresia (II.67) a regulatorului PID și modelul părții fixate (II.66) este următoarea:

$$G_0(s) = \frac{\frac{K_D K_a K_v}{\tau_v \tau_a} s^2 + \frac{K_P K_a K_v}{\tau_v \tau_a} s + \frac{K_I K_a K_v}{\tau_v \tau_a}}{s^4 + \frac{\tau_v + \tau_a}{\tau_v \tau_a} s^3 + \frac{1 + K_D K_a K_v}{\tau_v \tau_a} s^2 + \frac{K_P K_a K_v}{\tau_v \tau_a} s + \frac{K_I K_a K_v}{\tau_v \tau_a}} \quad (\text{II.68})$$

Polinomul caracteristic în buclă închisă are patru poli, dar doar trei dintre aceștia pot fi plasați și identificați cu parametrii de proiectare ai regulatorului PID, K_P , K_I și K_D . Al patrulea pol este determinat de coeficientul $\frac{\tau_v + \tau_a}{\tau_v \tau_a}$ [138]. Cealți trei poli sunt plasați în funcție de performanțele dorite (suprareglare și timp de răspuns). Egalând polinomul caracteristic obținut din performanțele impuse cu polinomul caracteristic în buclă închisă, rezultă:

$$\begin{aligned} & (s^2 + 0.6s + 0.1)(s + 0.4) \left(s + \frac{\tau_v + \tau_a}{\tau_v \tau_a} - 1 \right) \\ &= s^4 + \frac{\tau_v + \tau_a}{\tau_v \tau_a} s^3 + \left(\frac{\tau_v + \tau_a}{\tau_v \tau_a} - 0.66 \right) s^2 + \left(0.34 \frac{\tau_v + \tau_a}{\tau_v \tau_a} - 0.3 \right) s \\ &+ 0.04 \left(\frac{\tau_v + \tau_a}{\tau_v \tau_a} - 1 \right) \end{aligned} \quad (\text{II.69})$$

Parametrii regulatorului PID determinați din ecuația (II.93) sunt:

$$\begin{aligned} K_P &= \frac{0.34(\tau_v + \tau_a) - 0.3\tau_v\tau_a}{K_a K_v} \\ K_I &= \frac{0.04(\tau_v + \tau_a) - 0.04\tau_v\tau_a}{K_a K_v} \\ K_D &= \frac{\tau_v + \tau_a - 0.66\tau_v\tau_a - 1}{K_a K_v} \end{aligned} \quad (\text{II.70})$$

Regulatorul PID introduce două zerouri în funcția de transfer în buclă închisă care crește suprareglarea sistemului. Pentru a compensa efectul acestora a fost adăugat un bloc de anulare a zerourilor ca în Figura II.17 pe calea feed-forward. Acest bloc are următoarea formă:

$$G_{zc}(s) = \frac{\frac{K_I K_a K_v}{\tau_v \tau_a}}{\frac{K_D K_a K_v}{\tau_v \tau_a} s^2 + \frac{K_P K_a K_v}{\tau_v \tau_a} s + \frac{K_I K_a K_v}{\tau_v \tau_a}} \quad (\text{II.71})$$

II.2.4.4 Algoritm de control pentru urmăritori

Controlul distanțelor dintre vehicule d_i pentru vehiculele următoare i poate fi asigurat utilizând structura de control din Figura II.18, care conține un regulator GPC cu compensarea implicită a perturbațiilor pe calea feed-forward. Pentru a elimina efectele perturbației măsurabile (viteza v_{i-1} a vehiculului precedent), parametrul trebuie să fie zero [183]. Funcția de transfer a vehiculului $G_v(s)$ este utilizată cu forma din ecuația (II.64).

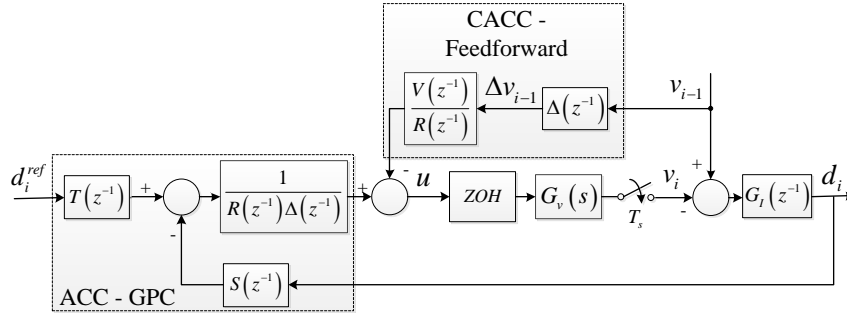


Figura II.18: Structura de control cu GPC a vehiculelor următoare

Luând în considerare forma discretă a funcției de transfer a vehiculelor (II.64) $G_v(z^{-1})$ și derivata ecuației distanței $\dot{d}_i = v_i - v_{i-1}$, modelul discret al părții fixe din Figura II.18 are următoarea formă:

$$y(k) = -G_I(z^{-1})G_v(z^{-1})u(k) + G_I(z^{-1})v_{i-1}(k) \quad (\text{II.72})$$

unde $G_I(z^{-1}) = \frac{z^{-1}T_s}{1-z^{-1}}$ este funcția de transfer discretă a integratorului și $G_v(z^{-1}) = \bar{B}(z^{-1})/\bar{A}(z^{-1})$.

Pornind de la modelul (II.72), poate fi construit modelul CARIMA:

$$A(z^{-1})y(k) = B(z^{-1})u(k-1) + D(z^{-1})v_{i-1}(k) + \frac{e(k)}{\Delta(z^{-1})} \quad (\text{II.73})$$

unde

$$\begin{aligned}
A(z^{-1}) &= \bar{A}(z^{-1})(1 - z^{-1}); \\
B(z^{-1}) &= -T_s \bar{B}(z^{-1}); \\
D(z^{-1}) &= z^{-1} T_s \bar{A}(z^{-1}); \\
\Delta(z^{-1}) &= 1 - z^{-1}.
\end{aligned}$$

Ecuția (II.73) poate fi rescrisă după cum urmează:

$$\tilde{A}(z^{-1})y(k) = B(z^{-1})\Delta u(k-1) + D(z^{-1})\Delta v_{i-1}(k) + e(k) \quad (\text{II.74})$$

unde $\tilde{A}(z^{-1}) = \Delta A(z^{-1})$.

Pentru a obține predictorul de ordin j , vor fi utilizate următoarele trei ecuații diofantice:

$$\begin{aligned}
&\bullet 1 = E_j(z^{-1})\tilde{A}(z^{-1}) + z^{-j}F_j(z^{-1}) \\
&\quad \begin{cases} E_j = 1 + \dots + e_{ne}z^{-ne}; & ne = j - 1 \\ F_j = f_0 + \dots + f_{nf}z^{-nf}; & nf = na \end{cases} \\
&\bullet E_j(z^{-1})B(z^{-1}) = G_j(z^{-1}) + z^{-j}H_j(z^{-1}) \\
&\quad \begin{cases} G_j(z^{-1}) = g_0 + g_1z^{-1} + \dots + g_{ng}z^{-ng}; & ng = j - 1 \\ H_j(z^{-1}) = h_0 + h_1z^{-1} + \dots + h_{nh}z^{-nh}; & nh = nb - 1 \end{cases} \\
&\bullet E_j(z^{-1})D(z^{-1}) = G'_j(z^{-1}) + z^{-j}H'_j(z^{-1}) \\
&\quad \begin{cases} G'_j(z^{-1}) = g'_0 + g'_1z^{-1} + \dots + g'_{ng}z^{-ng}; & ng = j - 1 \\ H'_j(z^{-1}) = h'_0 + h'_1z^{-1} + \dots + h'_{nh}z^{-nh}; & nh = nd - 1 \end{cases}
\end{aligned} \quad (\text{II.75})$$

rezultând astfel [184]:

$$\hat{y}(k+j|k) = G_j(z^{-1})\Delta u(k+j-1) + G'_j(z^{-1})\Delta v_{i-1}(k+j) + H_j(z^{-1})\Delta u(k-1) + H'_j(z^{-1})\Delta v_{i-1}(k) + F_j(z^{-1})y(k) \quad (\text{II.76})$$

Forma matriceală a predictorului (II.76) este:

$$\hat{\mathbf{y}} = \mathbf{G}\mathbf{u} + \mathbf{G}'\mathbf{v} + \mathbf{f} \quad (\text{II.77})$$

unde

$$\begin{aligned}
\hat{\mathbf{y}} &= \begin{bmatrix} \hat{y}(k+1|k) & \hat{y}(k+2|k) & \dots & \hat{y}(k+p|k) \end{bmatrix}^T \\
\mathbf{u} &= \begin{bmatrix} \Delta u(k|k) & \Delta u(k+1|k) & \dots & \Delta u(k+p-1|k) \end{bmatrix}^T \\
\mathbf{v} &= \begin{bmatrix} \Delta v(k+1|k) & \Delta v(k+2|k) & \dots & \Delta v(k+p|k) \end{bmatrix}^T \\
\mathbf{G} &= \begin{bmatrix} g_0 & 0 & \dots & 0 \\ g_1 & g_0 & \dots & 0 \\ \dots & \dots & \dots & \dots \\ g_{p-1} & g_{p-2} & \dots & g_0 \end{bmatrix} = \mathbf{G}' \\
\mathbf{f} &= \begin{bmatrix} f_{1.0} & \dots & f_{1.na} & h_{1.0} & \dots & h_{1.nb-1} \\ f_{2.0} & \dots & f_{2.na} & h_{2.0} & \dots & h_{2.nb-1} \\ \dots & \dots & \dots & \dots & \dots & \dots \\ f_{p.0} & \dots & f_{p.na} & h_{p.0} & \dots & h_{p.nb-1} \end{bmatrix} \begin{bmatrix} y(k) \\ \dots \\ y(k-na) \\ \Delta u(k-1) \\ \dots \\ \Delta u(k-nb) \end{bmatrix} \\
&+ \begin{bmatrix} h'_{1.0} & \dots & h'_{1.nd} \\ h'_{1.0} & \dots & h'_{1.nd} \\ \dots & \dots & \dots \\ h'_{1.0} & \dots & h'_{1.nd} \end{bmatrix} \begin{bmatrix} \Delta v_{i-1}(k) \\ \Delta v_{i-1}(k-1) \\ \dots \\ \Delta v_{i-1}(k-nd) \end{bmatrix}
\end{aligned} \quad (\text{II.78})$$

Funcția obiectiv este definită în [184] și considerând $v = 0$ din cauza valorilor foarte mici ale $\Delta v(k + j|k)$ peste orizontul de predicție p , rezultă:

$$J = (\hat{\mathbf{y}} - \mathbf{w})^T (\hat{\mathbf{y}} - \mathbf{w}) + \lambda \mathbf{u}^T \mathbf{u} = \frac{1}{2} \mathbf{u}^T \mathbf{H} \mathbf{u} + \mathbf{b}^T \mathbf{u} + \mathbf{f}_0 \quad (\text{II.79})$$

cu notațiile:

$$\begin{aligned} \mathbf{H} &= 2(\mathbf{G}^T \mathbf{G} + \lambda \mathbf{I}) \\ \mathbf{b}^T &= 2(\mathbf{f} - \mathbf{w})^T \mathbf{G} \\ \mathbf{f}_0 &= (\mathbf{f} - \mathbf{w})^T (\mathbf{f} - \mathbf{w}) \end{aligned}$$

Pentru a fi posibilă proiectarea regulatorului GPC, funcția obiectiv (II.79) trebuie derivată și egalată cu zero astfel:

$$\frac{\partial J}{\partial \mathbf{u}} = 0 \Rightarrow \mathbf{u} = \mathbf{H}^{-1} \mathbf{b} = (\mathbf{G}^T \mathbf{G} + \lambda \mathbf{I})^{-1} \mathbf{G}^T (\mathbf{w} - \mathbf{f}) \quad (\text{II.80})$$

Aplicând principiul orizontului recesiv, din relația (II.80) rezultă:

$$\begin{aligned} \Delta u(k|k) = \gamma(\mathbf{w} - \mathbf{f}) &= \sum_{j=1}^p \gamma_j w(k + j|k) - \sum_{j=1}^p \gamma_j F_j(z^{-1}) y(k) \\ &- \sum_{j=1}^p \gamma_j H_j(z^{-1}) \Delta u(k-1) - \sum_{j=1}^p \gamma_j H'_j(z^{-1}) \Delta v(k) \end{aligned} \quad (\text{II.81})$$

unde γ este prima linie a matricii: $(\mathbf{G}^T \mathbf{G} + \lambda \mathbf{I})^{-1} \mathbf{G}^T$.

Considerând ecuația (II.81), polinoamele regulatorului GPC pot fi determinate astfel:

$$\begin{aligned} R(z^{-1}) &= \left(1 + \sum_{j=1}^p \gamma_j z^{-1} H_j(z^{-1}) \right); \\ S(z^{-1}) &= \sum_{j=1}^p \gamma_j F_j(z^{-1}); \\ T(z^{-1}) &= \sum_{j=1}^p \gamma_j z^{-p+j}; \\ V(z^{-1}) &= \sum_{j=1}^p \gamma_j H'_j(z^{-1}). \end{aligned} \quad (\text{II.82})$$

În această situație, forma polinomială a regulatorului cu două grade de libertate și tratarea perturbațiilor măsurabile pe calea feedforward este:

$$\begin{aligned} R(z^{-1}) \Delta u(k|k) &= T(z^{-1}) w(k + p|k) - S(z^{-1}) y(k) \\ &- \underbrace{V(z^{-1}) \Delta v_{i-1}(k)}_{\text{feedforward}} \end{aligned} \quad (\text{II.83})$$

II.2.4.5 Proiectarea unui pluton cu CACC

Considerând algoritmi de control și structurile (Figura II.17 și Figura II.18) prezentate în cele două subsecțiuni precedente, a fost dezvoltată o diagramă de control pentru plutonul

ilustrat în Figura II.16 așa cum este ilustrată în Figura II.19, unde $G_{f_1}(s) = G_{f_2}(s) = \dots = G_{f_n}(s) = G_v(s)$ din relația (II.64).

Comunicațiile între vehicule oferă fiecărui vehicul posibilitatea de a transmite viteza succesorului său direct. Rețeaua wireless este modelată doar ca o legătură direct de la un vehicul la succesorul acestuia. Structurile de control ale următorilor primesc ca mărime de referință o distanță dependentă de viteză [185]:

$$d_i^{ref} = r_i + h_{d,i}v_i, \quad (II.84)$$

unde r_i este distanța dorită la staționare, $h_{d,i}$ este așa-numitul time-headway, și v_i este viteza vehiculului i ($i=1, \dots, n$). Time-headway este timpul necesar vehiculului i pentru a ajunge în poziția actuală a vehiculului precedent $i-1$ atunci când se deplasează cu o viteză constantă.

Utilizând metodele de control prezentate, liderul trebuie să fie capabil să-și mențină viteza în funcție de o referință, care este primită ca și intrare, iar următorii trebuie să se deplaseze în pluton la o distanță dorită față de vehiculul din fața acestuia calculată folosind viteza predecesorului direct introdusă ca o perturbație măsurabilă.

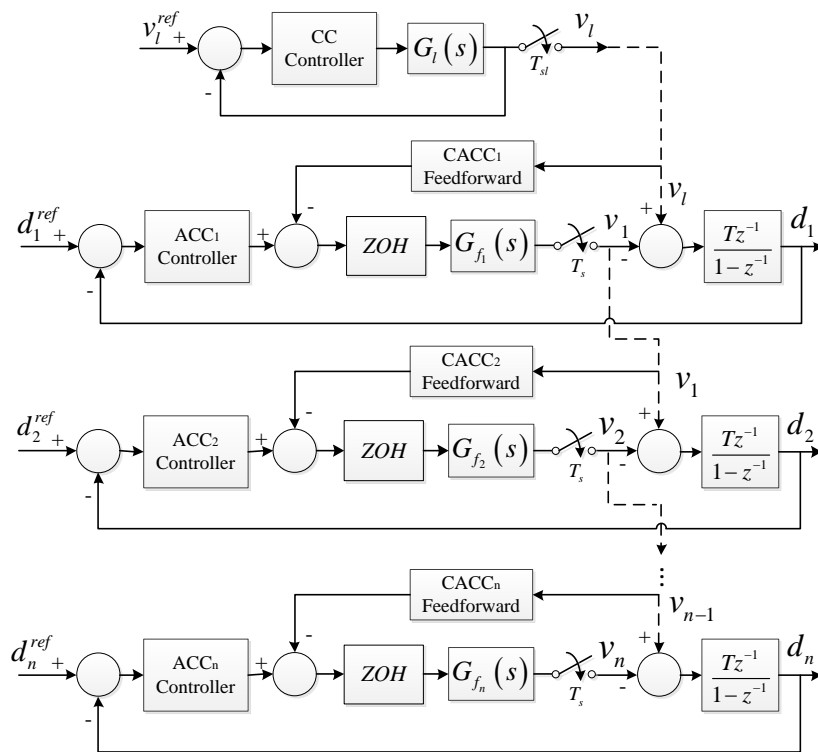


Figura II.19: Structura de control cu CACC a unui pluton cu n vehicule

II.2.4.6 Studiu de caz

Pentru testarea implementării sistemului VCPS proiectat, reprezentat de un pluton de vehicule cu CACC, a fost considerat cazul specific al unui pluton cu un lider și 15 urmăritori,

toate vehiculele conținând aceeași parametri ai modelului:

$$\begin{aligned}
 g &= 9.81\text{m/s}, v_0 = 20\text{m/s}, \theta_0 = 0, m = 1000\text{kg}, \\
 \rho &= 1.202\text{kg/m}^3, A = 1.5\text{m}^2, C_d = 0.5, \\
 f &= 0.015, v_w = 2\text{m/s},
 \end{aligned}
 \tag{II.85}$$

utilizate în funcția de transfer a modelului vehiculului din (II.66) pentru lider și din (II.64) pentru urmăritori. Simularea a fost realizată folosind mediul software Matlab/Simulink.

Toate datele din structurile Simulink utilizate sunt calculate și transmise la o perioadă de eșantionare de $T_s = 20\text{ms}$. Parametrii K_v și τ_v , valabile pentru toate vehiculele din pluton au fost calculate utilizând relațiile (II.63). Pentru liderul plutonului, parametrii elementului de execuție considerați în modelul vehiculului sunt $K_a = 10$ și $\tau_a = 0.2\text{s}$. Nu au fost considerate alte întârzieri suplimentare.

În Figura II.20, este ilustrată diagrama Simulink a plutonului bazat pe VCPS, unde se poate observa că fiecare urmăritor primește ca intrări distanța de referință și viteze vehiculului precedent și returnează distanța dintre acesta și predecesorul lui. Viteza controlată și comanda generată de regulatorul GPC sunt extrase doar pentru măsurători. Liderul prezintă la intrare un profil de viteză în km/h și la ieșire viteza controlată de către sistemul CC proiectat.

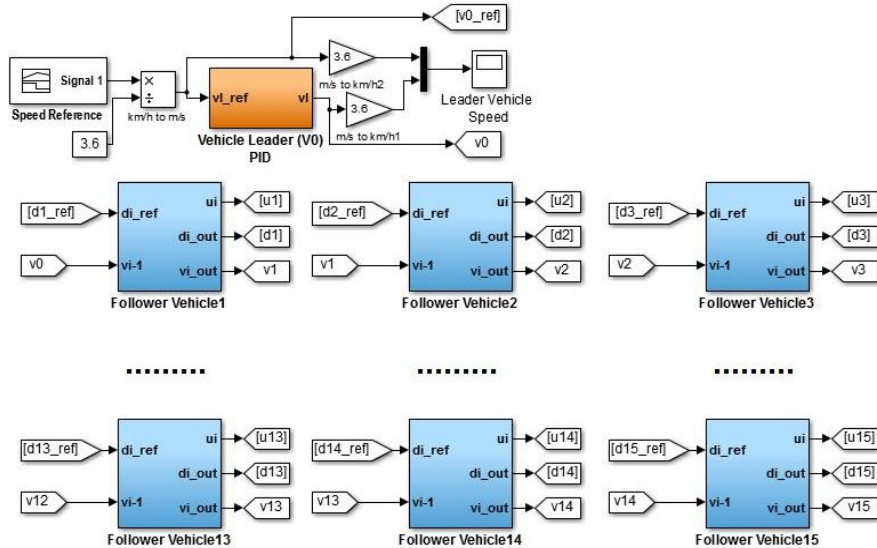


Figura II.20: Reprezentarea Simulink a plutonului bazat pe VCPS cu un lider și 15 urmăritori

După proiectarea regulatorului PID, care ajută liderul să urmărească o anumită referință de viteză, parametrii regulatorului sunt: $K_P = 28.0899$, $K_I = 3.2106$ și $K_D = 85.7422$. Considerând că regulatorul PID introduce două zerouri în sistem pentru a reduce efectele acestora a fost proiectat un bloc de anulare a zerourilor.

Referința de distanță, primită de urmăritori ca și intrare, este compusă dintr-o distanță de staționare $r_i = 1\text{m}$, un timp de înaintare (time-headway) $h_{d,i} = 100\text{ms}$ și din viteza vehiculului v_i . Vehiculele urmăriitoare conțin un regulator GPC dedicat cu tratarea perturbațiilor

măsurabile pe calea feed-forward care asigură menținerea unei anumite distanțe între vehicule dependentă de viteză.

În studiul de caz prezent, perturbația măsurabilă este viteza vehiculului precedent, care este transmisă direct către vehiculul controlat. Considerarea comunicațiilor între vehicule face posibilă transmisia datelor de la un vehicul la altul. Pentru a asigura performanțele regulatorului factorul de ponderare este $\lambda = 0$ și orizontul de predicție este $h_p = 4$.

În Figura II.21, sunt ilustrate vitezele vehiculelor din pluton, inclusiv referința de viteză a liderului. Mărind diagrama în două zone diferite se poate observa că aceste semnale de viteză urmăresc referința de viteză a liderului plutonului, prezentând erori foarte mici când referința este un semnal de tip rampă (Figura II.21 zoom-ul din stânga) și nicio eroare când viteza de referință este constantă (Figura II.21 zoom-ul din dreapta). Performanța sistemului ACC-GPC poate fi demonstrată analizând semnalele din Figura II.22, care arată distanțele între vehicule pentru întregul pluton. Aceste forme de undă au aceeași formă ca și vitezele din Figura II.21 deoarece referințele pentru distanțe sunt dependente de viteză. Ca și în cazul precedent, dacă se măresc anumite zone de pe grafic, se pot observa diferențe extrem de mici (mai puțin de 10 mm) între distanțe care apar la viteze variabile (Figura II.22 zoom-ul din stânga), dar se obține aceeași distanță la viteze constante (Figura II.22 zoom-ul din dreapta).

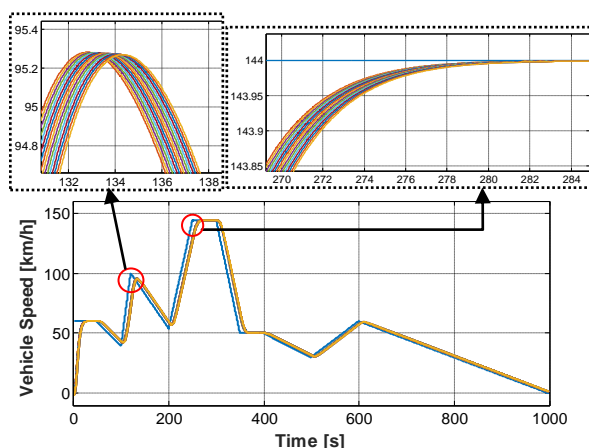


Figura II.21: Vitezele vehiculelor din pluton

Timpul de răspuns făcând referire la două semnale consecutive are o valoare de aproximativ 10 ms, iar suprareglarea este foarte mică astfel încât nu este vizibilă în niciuna dintre ultimele două figuri ilustrate. Răspunsurile obținute dovedesc că toate vehiculele din pluton sunt capabile să mențină aceeași distanță față de vehiculul din față la un anumit timp. Acestea reacționează la accelerările și decelerările predecesorilor predictând evoluția vitezelor lor. Aceste performanțe asigură faptul că plutonul experimental proiectat este stabil considerând că doar viteza predecesorilor este aplicată ca și perturbație cunoscută.

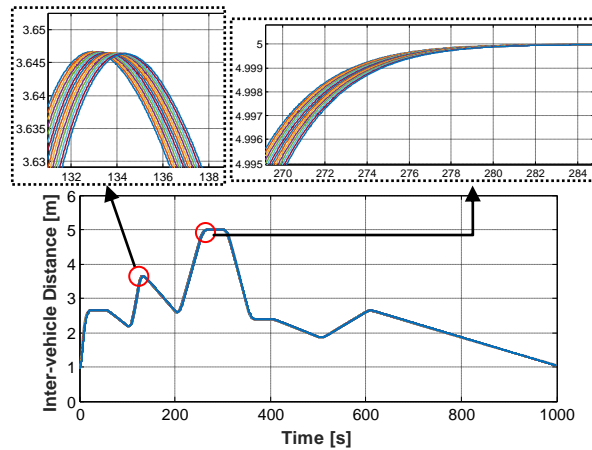


Figura II.22: Distanțele dintre vehicule considerând o referință dependentă de viteză

II.2.5 Controlul cooperativ al unui pluton de vehicule pentru creșterea capacității arte-relor de circulație urbane

O soluție pentru maximizarea eficienței capacității sistemelor de transport cu reducerea aglomerației urbane și pentru îmbunătățirea siguranței traficului este proiectarea și implementarea sistemelor inteligente de transport. Aceste sisteme inteligente sunt bazate pe introducerea automatizării în vehicule și comunicațiile wireless într-un mediu cu vehicule conectate [185]. Primul sistem automat de asistență a șoferului a fost sistemul de control a vitezei de croazieră (cruise control CC) care permite conducerea vehiculelor cu o viteză dorită. Sistemul adaptiv pentru controlul vitezei de croazieră (ACC) a fost următorul pas al automatizării care îmbunătățește CC prin adăugarea posibilității de a conduce un vehicul în spatele unui lider la o anumită distanță, care îmbunătățește capacitatea drumului și siguranța traficului precum și eficiența consumului de combustibil. Mai târziu, aplicațiile sistemului ACC au fost extinse pentru a forma un pluton cunoscut ca și sistemul cooperativ adaptiv de control al vitezei de croazieră (CACC).

Conceptul CACC este bazat pe combinarea controlului automat al vitezei/distanței cu un element cooperativ, cum sunt comunicațiile între vehicule (V2V) și/sau între infrastructură și vehicul (I2V) [186]. Comunicațiile V2V furnizează informații despre vehiculul precedent, iar comunicațiile I2V oferă informații despre limitele de viteză variabile sau pornirea coordonată a vehiculelor care așteaptă la culoarea roșie a semaforului care pot accelera într-un mod coordonat în momentul în care semnalul semaforului devine verde. Vehiculele echipate cu CACC pot folosi sistemul doar când urmăresc alte vehicule echipate cu astfel de sisteme. Capabilitățile sistemului CACC suplimentate de informațiile legate de faza de semnal și de sincronizare (SPAT) de la infrastructură pot aduce beneficii în intersecții prin informarea șoferilor cu privire la cea mai eficientă viteză cu care să se apropie și să treacă prin intersecție [187]. Mediul unei artere de circulație este mult mai dinamic decât o autostradă clasică, deoarece include intersecții, o largă varietate de manevre ale

vehiculelor, căi comerciale și rezidențiale, și pietoni. Beneficiile comunicațiilor V2V și I2V ar putea include o deplasare mult mai eficientă a vehiculelor printr-o secțiune arterială, economisind timpul de călătorie și reducând utilizarea combustibilului și eliminarea emisiilor. O intersecție cu semafor de culoare roșie cu informații SPAT ar putea fi folosită de infrastructură să determine viteza cea mai potrivită cu care vehiculele echipate cu sisteme CACC să poată trece cât mai repede posibil prin intersecție fără să oprească. Tehnologia CACC ar putea de asemenea să activeze beneficiile siguranței de-a lungul arterelor de circulație. Comunicațiile I2V ar putea avertiza șoferii cu privire la vehiculele cu care se pot intersecta, detectarea trecerilor de pietoni sau apropierea vehiculelor speciale pentru intervenții de urgență.

Acest capitol se axează pe conceptul de pornire coordonată pe arterele de circulație, unde vehiculele cu CACC care așteaptă la un semnal de culoare roșie al semaforului formează un pluton, care este capabil să accelereze într-o manieră coordonată în momentul în care semnalul semaforului se schimbă în verde. Această pornire coordonată ar putea permite mai multor vehicule să treacă printr-o intersecție aglomerată într-un ciclu de verde decât în cazul conducerii manuale. Sistemul CACC pentru un pluton de vehicule utilizează comunicații doar cu vehiculul imediat precedent care sunt implementate ca un semnal feed-forward. Principalul obiectiv de control pentru un pluton de vehicule este să urmărească vehiculul precedent la o distanță dorită care depinde de viteza vehiculului. Se presupune că liderul plutonului urmărește o referință de viteză variabilă în timp cu ajutorul sistemului CC de când culoarea semaforului se schimbă în verde. Celelalte vehicule ale plutonului folosesc dispozitive radar pentru a măsura distanța dintre acestea și vehiculul din fața lor, care sunt folosite în regulatoare ACC standard de tip feedback. Viteza vehiculului precedent este valabilă prin intermediul comunicațiilor wireless și este utilizată de un regulator feed-forward. Proiectarea sistemului CACC a fost testată cu succes în simulare.

II.2.5.1 Modelarea vehiculelor din pluton

Prin organizarea vehiculelor în plutoane capacitatea unei intersecții poate fi crescută utilizând modelarea adecvată și strategii de control ale vehiculelor din pluton. Este considerată o intersecție cu patru drumuri și semafor așa cum este ilustrată în Figura II.23 și se presupune că vehiculele se pot organiza într-un pluton atunci când străbat intersecția. Pentru a forma un pluton, vehiculele trebuie să aibă capabilități CACC care funcționează când vehiculul este oprit, astfel încât atunci când semaforul se schimbă în verde, toate vehiculele din coadă se deplasează împreună cu un timp de deplasare mic între ele. Toți șoferii trebuie să activeze modul CACC când opresc într-o intersecție. Formarea plutonului poate fi făcută de asemenea cu vehicule care dețin capabilități ACC, dar timpul de înaintare va fi mai mare.

Conceptul CACC este bazat pe combinarea controlului automat al vitezei și controlul distanței cu un element cooperativ, cum sunt comunicațiile V2V și/sau I2V. Vehiculele echipate cu CACC pot folosi sistemul doar când urmăresc alte vehicule echipate cu acest sistem.

Pentru controlul automat al vitezei, conceptul CC este utilizat pentru liderul plutonului. Primul vehicul al plutonului este de obicei modelat pe baza ecuației de mișcare longitudi-

nală a vehiculului [138]:

$$m \frac{dv}{dt} = F_x - mg \sin \theta - fmg \cos \theta - 0.5\rho AC_d(v + v_w)^2 \quad (\text{II.86})$$

unde m este masa vehiculului, F_x este forța de tracțiune, v este viteza vehiculului, v_w este viteza vântului, g este accelerația gravitațională, θ este înclinarea drumului, ρ este densitatea aerului, C_d este coeficientul de frecare, f este coeficientul de rezistență la rulare, și A este aria frontală a vehiculului. Liniarizând ecuația (II.86) prin considerarea unui punct nominal de funcționare cu v_0 și θ_0 , se obține un model simplu cu o singură variabilă de stare:

$$\dot{x} = -\frac{1}{\tau_v}x + \frac{K_v}{\tau_v}u + w \quad (\text{II.87})$$

unde $x = \Delta v$, $u = \Delta F_x$ și:

$$\begin{cases} \tau_v = m / (\rho C_d A (v_0 + v_w)) \\ K_v = 1 / (\rho C_d A (v_0 + v_w)) \\ w = mg (f \sin \theta_0 - \cos \theta_0) \Delta \theta \end{cases}$$

Pentru controlul automat al distanței, conceptul ACC este utilizat pentru vehiculele urmăriitoare. Principalul obiectiv pentru fiecare următor cu respectarea siguranței în trafic este menținerea unei distanțe prestabilite față de vehiculul precedent din pluton. Un vehicul următor este cel mai adesea modelat în literatură [138] ca un sistem dinamic cu două stări: x_1 - distanța dintre vehiculul controlat și vehiculul precedent și x_2 - viteza vehiculului controlat.

$$\begin{cases} \dot{x}_1 = -x_2 + v_p \\ \dot{x}_2 = -\frac{1}{\tau_v}x_2 + \frac{K_v}{\tau_v}u + w \end{cases} \Leftrightarrow \begin{cases} \dot{x}_v = A_c x_v + b_c u + d_w w + d_v v_p \\ y = [1 \ 0] \end{cases} \quad (\text{II.88})$$

Viteza vehiculului precedent v_p nu este măsurabilă și este considerată o perturbație de intrare, care poate fi modelată cu un semnal de tip rampă. Viteza controlată nu poate urmări viteza vehiculului precedent nici măcar în cazul manevrelor de accelerare dacă nu este adăugat un integrator cu două stări la sistemul dinamic (II.88):

$$\begin{cases} \dot{x}_3 = r - x_1 \\ \dot{x}_4 = x_3 \end{cases} \Leftrightarrow \{\dot{x}_i = A_i x_i + h r\} \quad (\text{II.89})$$

unde r este referința impusă pentru distanță, x_3 este integrala erorii și x_4 este integrala dublă a erorii.

Combinând (II.88) și (II.89), modelul augmentat al unui vehicul următor controlat devine:

$$\begin{cases} \dot{x} = A_a x + b_a u + d_{aw} w + d_{av} v_p + h_a r \\ y = [1 \ 0 \ 0 \ 0] x \end{cases} \quad (\text{II.90})$$

unde $x = [x_v \ x_i]^T$. Matricile și vectorii din ecuația (II.90) pot fi calculate ușor din (II.88) și (II.89).

Dacă este disponibil un sistem CACC, viteza vehiculului precedent poate fi transmisă către vehiculul controlat utilizând comunicațiile V2V, și astfel este transformată într-o perturbație măsurabilă în modelul (II.90). Utilizând un regulator feed-forward adecvat,

pertur-bația măsurabilă v_p poate fi eliminată. Dacă sistemul CACC deține și comunicații I2V prin care fazele semaforului sunt transmise către liderul plutonului, atunci vehiculele din pluton care așteaptă la culoarea roșie a semaforului vor începe să accelereze într-un mod coordonat în momentul în care semaforul devine verde.

II.2.5.2 Sistemul CACC pentru controlul longitudinal al unui pluton de vehicule

Este considerat un pluton de vehicule cu capabilități CACC format într-o intersecție când semnalul semaforului a fost roșu și care așteaptă culoarea verde a semaforului (Figura II.23). Distanța dintre vehicule este d_0 , distanța dorită la staționare, măsurată cu un senzor de tip radar. Când semnalul semaforului devine verde, liderul plutonului primește această informație prin intermediul sistemului de comunicații I2V și va accelera pentru a atinge viteza maximă admisibilă cât mai repede posibil. Vehiculele urmăriitoare din pluton vor accelera în același timp cu liderul, menținând o distanță de siguranță între vehicule care va reprezenta referința pentru sistemul de control al distanței:

$$r = d_0 + t_h v_p \quad (\text{II.91})$$

unde t_h este timpul de înaintare dorit. Referința este calculată pentru motive de siguranță în funcție de viteza vehiculului precedent v_p , obținută prin sistemul de comunicații V2V. Sistemul CACC este proiectat utilizând conceptul CC pentru lider și ACC pentru urmăritori cu o componentă feed-forward pentru eliminarea efectului perturbației măsurabile dat de viteza vehiculului precedent.

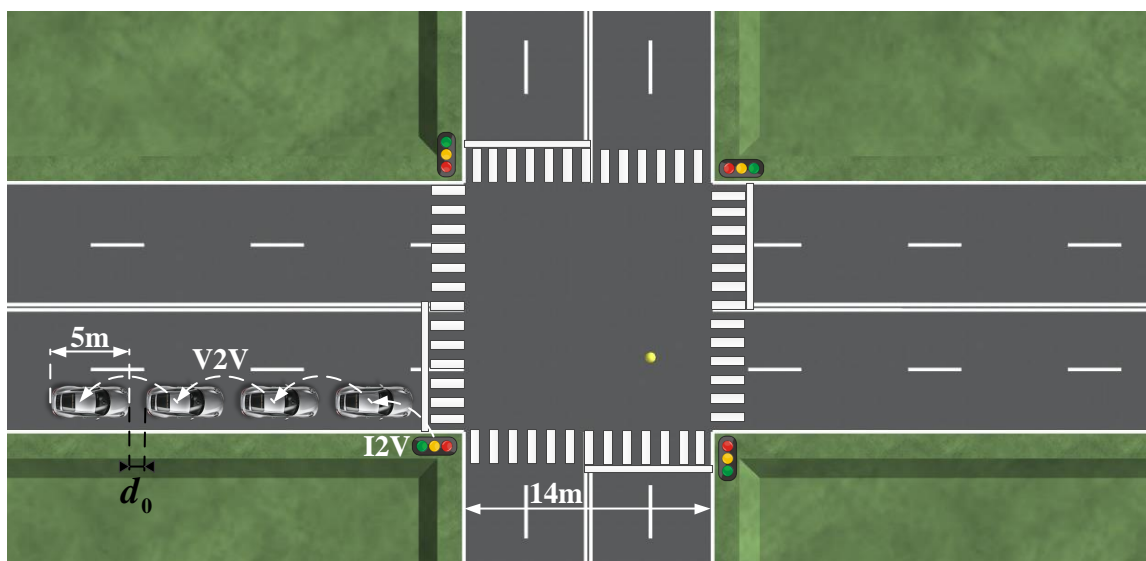


Figura II.23: Plutonul de vehicule într-o intersecție de patru drumuri cu semafor

În Figura II.24 este ilustrat plutonul cu CACC descris anterior, în care p_i ($i = \overline{1, n}$) este poziția vehiculelor urmăriitoare, p_l este poziția liderului și L_{veh} reprezintă lungimea vehiculelor. Se presupune că toate vehiculele au aceeași lungime. Comparativ cu Figura II.23, în Figura II.24 plutonul este reprezentat ca fiind în mișcare.

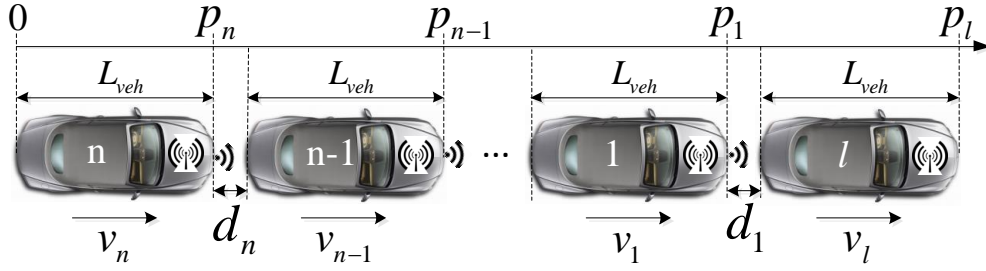


Figura II.24: Un pluton de vehicule format cât timp așteaptă semnalul verde al semaforului (în mișcare)

A. Proiectarea sistemului CC Liderul plutonului trebuie să accelereze când primește informația de la sistemul de comunicații I2V că semaforul a devenit verde și să atingă în cât mai scurt timp posibil viteza maximă admisibilă. Această cerință poate fi îndeplinită utilizând conceptul CC. Regulatorul pentru sistemul CC ca fi proiectat pe baza modelului de mișcare longitudinală a vehiculului (II.87) și considerând ca referință viteza maximă admisibilă. Convertind modelul (II.87) în domeniul discret, se obține funcția de transfer discretă:

$$G_l(z^{-1}) = \frac{b_1 z^{-1}}{1 + a_1 z^{-1}} \quad (\text{II.92})$$

Utilizând (II.92), poate fi proiectat un regulator discret PI cu parametrii de proiectare $q_0 = K_p(1 + T_s/T_i)$ și $q_1 = -K_p T_s/T_i$ prin metoda alocării polilor. Luând în considerare performanțele (suprareglare lipsă și un timp de răspuns scurt) parametrii de proiectare pot fi proiectați prin manipularea polilor polinomului caracteristic în buclă închisă de ordin doi cu factorul de amortizare ζ și frecvența naturală ω_n :

$$\begin{aligned} P_c(z^{-1}) &= 1 + \alpha_1 z^{-1} + \alpha_2 z^{-2}, \text{ cu :} \\ \alpha_1 &= -2e^{-\zeta\omega_n T_s} \cos(\omega_n T_s \sqrt{1 - \zeta^2}); \alpha_2 = e^{-2\zeta\omega_n T_s} \end{aligned} \quad (\text{II.93})$$

rezultând:

$$q_0 = (\alpha_1 - a_1 + 1)/b_1; q_1 = (\alpha_2 + a_1)/b_1 \quad (\text{II.94})$$

Regulatorul PI introduce un zero la funcția de transfer în buclă închisă care ar putea duce la creșterea suprareglării sistemului CC. Zero-ul introdus de regulatorul PI poate fi compensat prin introducerea unui bloc de anulare a zeroului pe calea feedforward a referinței cu funcția de transfer:

$$G_{zc}(z^{-1}) = \frac{(1 + \alpha_1 + \alpha_2)z^{-1}}{b_1 q_0 + b_1 q_1 z^{-1}} \quad (\text{II.95})$$

B. Proiectarea sistemului ACC Vehiculele următoare trebuie să urmărească liderul, menținând o distanță de siguranță (II.91) între vehicule. Sistemul de control al fiecărui vehicul are două regulatoare: un regulator cu reacție negativă pentru menținerea distanțelor dorite dintre vehicule și unul feed-forward pentru a compensa efectul perturbației măsurabile. Distanța dorită este menținută de un regulator ACC. Considerând posibilitatea apariției unei erori în sistemul de comunicații wireless, regulatorul ACC trebuie să aibă un dublu integrator al erorii deoarece se presupune că viteza vehiculului precedent, care acționează ca o

perturbație, poate fi modelată ca un semnal de tip rampă. Ca un rezultat, vehiculul controlat poate urmări vehiculul precedent chiar și în cazul accelerațiilor și decelerațiilor mari. Necesitatea unui dublu integrator a făcut imposibilă utilizarea unui regulator PID. Considerând aceasta, a fost folosit un algoritm de control discret bazat pe stări cu dublu integrator pentru componenta feedback a regulatorului ACC. Componenta feed-forward a regulatorului ACC este proiectată pentru a elimina perturbația măsurabilă generată de viteza vehiculului din față. Astfel, legea de control cu cele două componente este dată de:

$$u = u_{fb} + u_{ff} = Kx + G_{ff}v_p \quad (\text{II.96})$$

Proiectarea regulatorului de stare este bazată pe modelul (II.90) cu presupunerea că perturbația măsurabilă v_p a fost eliminată de regulatorul feed-forward. Într-un final, a fost obținut următorul model liniar discret din (II.90) utilizând perioada de eșantionare T_s și considerând $\Delta\theta = 0$:

$$\begin{cases} x(k+1) = A_d x(k) + b_d u(k) + h_a r(k) \\ y(k) = [1 \ 0 \ 0 \ 0] x(k) \end{cases} \quad (\text{II.97})$$

unde $A_d = I_4 + T_s A_a$ și $b_d = b_a T_s + A_a b_a T_s^2 / 2$. Pentru modelul discret de stare (II.97), legea de control $u_{fb}(k) = Kx(k) = [K_x \ K_i] [x_v(k) \ x_i(k)]^T$ este proiectată utilizând formula Ackermann:

$$K = [0 \ 0 \ 0 \ 1] R^{-1} P_c(A_d) \quad (\text{II.98})$$

unde R este matricea de controlabilitate și P_c este polinomul caracteristic al sistemului de control în buclă închisă.

Regulatorul feed-forward este proiectat prin considerarea funcției de transfer în buclă închisă a perturbației. Utilizând modelul (II.90) cu $w = 0$ ($\Delta\theta = 0$) și legea de control (II.96), se obține modelul în buclă închisă:

$$\begin{cases} \dot{x} = (A_a + b_a K)x + (b_a G_{ff} + d_{av})v_p + h_a r \\ y = c^T x \end{cases} \quad (\text{II.99})$$

Aplicând transformata Laplace asupra ecuațiilor (II.99) cu $r = 0$, rezultă funcția de transfer în buclă închisă a perturbației $G_{0d}(s)$:

$$Y(s) = \underbrace{c^T (Is - (A_a + b_a K))^{-1} (b_a G_{ff} + d_{av})}_{G_{0d}(s)} V_p(s) \quad (\text{II.100})$$

Pentru a obține eliminarea perturbației măsurabile v_p , este necesară anularea funcției de transfer a perturbației $G_{0d}(s)$. Astfel, se obține un regulator feed-forward PD cu următoarea funcție de transfer:

$$G_{ff}(s) = K_p \frac{sT_d + 1}{sT_f + 1} \quad (\text{II.101})$$

$$cu : K_p = \frac{1 - k_2 K_v}{K_v}; T_d = \frac{\tau_v}{1 - k_2 K_v}; T_f = T_d / N_f$$

unde $k_2 = K_x(2)$ și $N_f = 8 \div 20$.

Pentru implementarea regulatorului feed-forward va fi folosită forma discretă a funcției de transfer (II.101).

II.2.5.3 Studiu de caz

Sistemul CACC dezvoltat pentru controlul longitudinal al unui pluton de vehicule cu scopul de a crește capacitatea arterelor urbane de circulație este reprezentat în Figura II.25 și este utilizat pentru ca vehiculele să accelereze într-un mod coordonat când semnalul semaforului devine verde. În acest mod, vor trece prin intersecție mai multe vehicule într-un ciclu de verde al semaforului decât atunci când vehiculele sunt conduse manual. În studiul realizat, vehiculele următoare cu CACC folosesc sisteme de comunicații V2V pentru a primi mesaje trimise de către vehiculul imediat precedent și liderul utilizează un sistem de comunicații I2V pentru a recepționa informații cu privire la faza semaforului. Toate vehiculele sunt echipate cu dispozitive radar pentru a măsura distanța față de vehiculul precedent.

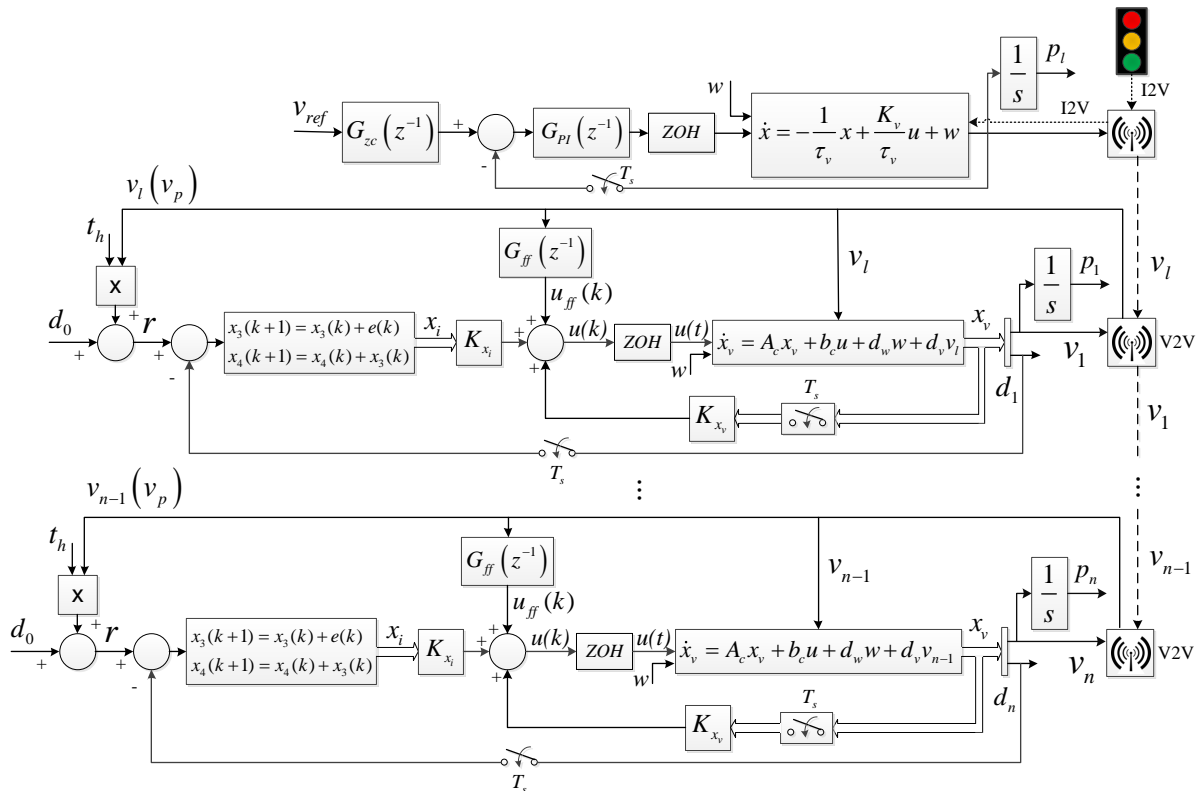


Figura II.25: Structura unui pluton cu n vehicule format într-o intersecție)

Este considerată intersecția din Figura II.23 cu patru drumuri, fiecare drum având o bandă de mers înainte și alta pentru virajul la stânga. În [188] capacitatea unei intersecții este definită ca:

$$C = \sum_i s_i \frac{g_i}{T_s} \quad (II.102)$$

unde T_s este perioada de eșantionare, pentru grupul i , s_i este rata de saturare a fluxului de vehicule și g_i/T_s este raportul efectiv pentru culoarea verde a semaforului. Pentru rata de

saturare, în [188] este dată următoarea formulă:

$$s_i = N s_0 f \quad (\text{II.103})$$

unde N este numărul de benzi dintr-un grup, s_0 este rata de bază măsurată în vehicule pe oră (vph), și f este un factor de ajustare care ia în considerare geometria drumului și natura traficului. În [189], pentru o reprezentare similară ca cea din Figura II.23, se presupune că fiecare bandă suportă un flux de până la $s_0 = 1900$ vph, rezultând o capacitate total de $C = 1900 \cdot 8 = 15200$ vph calculată cu (II.102) și considerând $f = 1$. Din Figura II.23 rezultă că doar două mișcări pot fi permise în același timp cu respectarea siguranței, astfel rata de verde pentru fiecare mișcare este cel mult $g_i/T_s = 0.25$, și din ecuația (II.102) capacitatea intersecției scade la 3800 vph. Intersecția este principalul lucru care conduce la blocaje în trafic pe drumurile urbane: capacitatea acesteia este o fracțiune (aici 1/4) din capacitatea drumurilor conectate cu aceasta.

Acest studiu explorează utilizarea sistemului CACC pentru a crește capacitatea arterelor urbane bazată pe organizarea vehiculelor într-un pluton și accelerarea acestora într-o manieră coordonată când culoarea semaforului devine verde. Se consideră un pluton de 16 vehicule (un lider și $n = 15$ urmăritori) având pozițiile din Figura II.24, cât timp așteaptă la culoarea roșie a semaforului. Parametrii plutonului sunt dați în Tabelul 5.

| Parametru | Valoare |
|--|-----------------------------|
| Distanța minimă la staționare (d_0) | 2 m |
| Țimpul de înaintare de siguranță (t_h) | 0.7 s |
| Țimpul de înaintare pentru vehiculele dinafara plutonului (t_{hout}) | 1.8 s |
| Accelerația maximă | 0.75 m/s ² |
| Lungimea vehiculului (Lveh) | 5 m |
| Viteza dorită (vref) | 50 km/h (≈ 14 m/s) |

Table 5: Parametrii plutonului

Pentru vehiculele conectate din pluton, se consideră un timp de înaintare mic (0.7s), care asigură proprietatea de stabilitate a șirului [190] pentru sistemul controlat CACC, iar pentru vehiculele conduse manual se consideră un timp mai mare (1.8s) [191]. Considerând limita de viteză în intersecție de 50 km/h, viteza de referință pentru liderul vehiculului a fost aleasă egală cu această limită.

Structura sistemului CACC utilizat pentru a controla plutonul este reprezentată în Figura II.25. Liderul primește fazele semaforului prin sistemul de comunicații I2V și transmite viteza acestuia către primul vehicul urmăritor prin comunicații V2V. Fiecare urmăritor primește/transmite viteza de la predecesorul/către succesorul acestuia prin sistemul de comunicații V2V. Această informație este utilizată pentru a calcula referința r pentru sistemele de control a distanței cu (II.91) și de a anula perturbația cu regulatorul feed-forward, cum se poate observa în Figura II.25. Pentru fiecare vehicul din pluton, se calculează și poziția acestora, rezultând p_i și p_i , $i = \overline{1, n}$.

Proiectarea reguletoarelor CACC este bazată pe modelele (II.87) și (II.90) cu parametrii $K_v = 0.075$ și $\tau_v = 75.6$ s din [138]. Ca și performanțe, pentru sistemele CACC suprareglarea nu este prezentă, iar durata regimului tranzitoriu a sistemului ACC a fost calculată folosind

acelerația maximă din Tabelul 5. Pentru sistemul CC a fost determinat un timp de regim tranzitoriu mai mic decât pentru sistemele ACC.

Regulatorul discret PI al sistemului CC a fost proiectat utilizând metoda plasării polilor din secțiunea precedentă, rezultând următoarele valori pentru parametrii de proiectare: $q_0 = 378.3115$ și $q_1 = -377.8420$ pentru o perioadă de eșantionare $T_s = 0.01s$. Lipsa suprareglării a fost furnizată de blocul de anulare a zeroului cu funcția de transfer (II.95).

Sistemul ACC conține două regulatoare. Pentru regulatorul discret de stare, matricea de stare obținută este $K = [k_1 k_2 k_3 k_4]$, iar pentru regulatorul feed-forward PD au fost determinați următorii parametri: $K_p = 17.1511$, $T_d = 0.2204$ și $T_f = 0.0110$.

Simulatorul plutonului bazat pe schema bloc din Figura II.25 este inițializat când se începe cu pozițiile de start ale vehiculelor:

$$\begin{aligned} p_l^0 &= (n + 1)L_{veh} + nd_0 \\ p_i^0 &= (n - i + 1)L_{veh} + (n - i)d_0, i = \overline{1, n}. \end{aligned} \quad (II.104)$$

De asemenea, integratoarele care calculează distanțele sunt inițializate cu d_0 , distanța minimă la staționare. Datorită inițializărilor, experimentul de simulare are două faze:

- în primele 40 de secunde, se consideră $v_{ref} = 0$ pentru a obține starea de echilibru a plutonului cu valorile inițiale;
- la $t = 40s$ culoarea semaforului devine verde, iar referința de viteză este considerată valoarea maximă admisă $v_{ref} = 50$ km/h.

Simularea furnizează ca și rezultat distanțele între vehicule ilustrate în Figura II.26, care sunt controlate prin regulatorul de stare ACC și de regulatorul feed-forward PD pentru a menține o anumită distanță între vehiculele din pluton. În acest caz a fost considerat un timp de înaintare $t_h = 0.7s$.

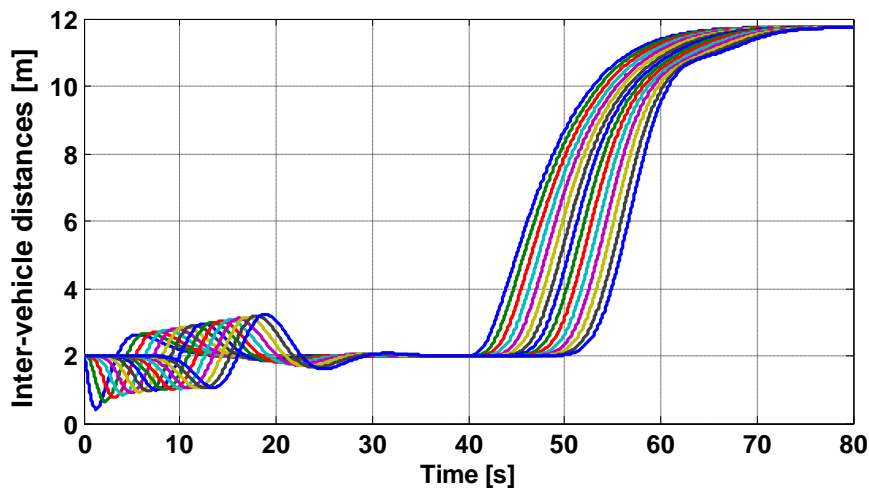


Figura II.26: Distanțele dintre vehiculele din pluton (cu timpul de înaintare t_h)

Figura II.26 ilustrează distanțele între vehicule obținute prin calcularea în timpul deplasării prin intersecție a vehiculelor care nu sunt organizate într-un pluton. Acestea sunt reprezentate de un timp de deplasare $t_{hout} = 1.8s$.

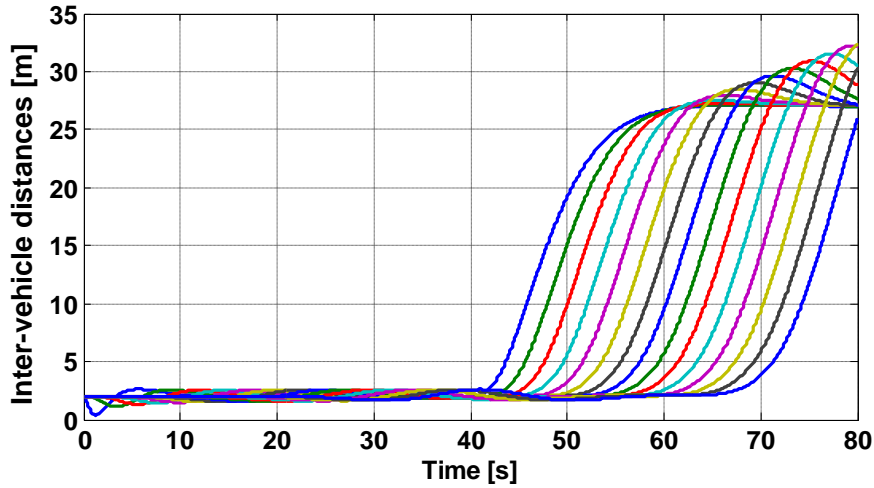


Figura II.27: Distanțele dintre vehiculele din pluton (cu timpul de înaintare $t_{h,out}$)

Presupunând că durata în care semaforul este verde este egală cu 30s în Figura II.27 se poate observa că toate cele 16 vehicule din pluton au trecut cu succes prin intersecție.

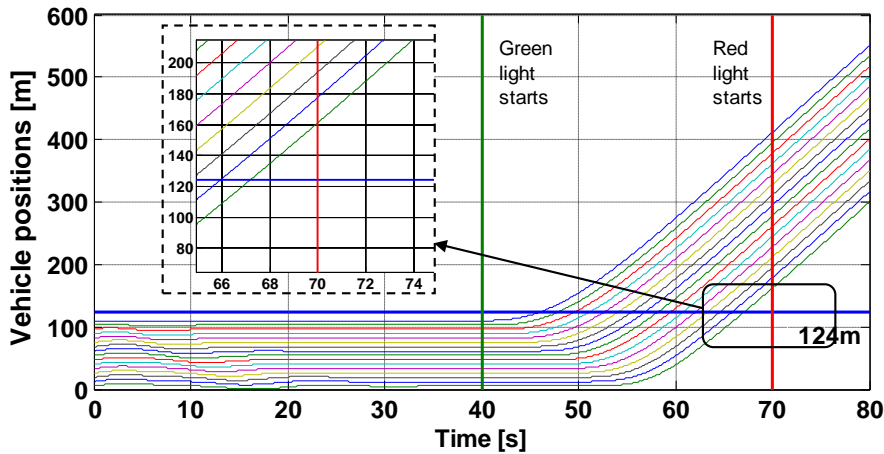


Figura II.28: Pozițiile vehiculelor din pluton

Întregul pluton trebuie să se deplaseze cel puțin 124m (lungimea plutonului: $16 * L_{veh} + 15 * d_0 = 110m$, la care se adaugă lățimea drumului perpendicular pe drumul pe care se deplasează vehiculele: 14m) pentru a trece cu succes prin intersecție. Pozițiile vehiculelor sunt controlate așa cum se așteaptă și astfel alinierea plutonului este stabilă. Considerând cazul conducerii manuale, timpul de deplasare al vehiculelor este setat la $t_{h,out}$ și astfel doar 9 vehicule din 16 au destul timp să treacă prin intersecție în 30s.

II.2.6 Controlul predictiv al dinamicii vehiculelor autonome

În acest moment, cercetările desfășurate în direcția dezvoltării vehiculelor autonome, indică spre strategiile de control predictiv, ca fiind cele mai potrivite metode pentru controlul traiectoriei acestor vehicule. Aplicând controlul predictiv, la orice moment de timp, comenzile aplicate (virare, frânare, accelerare) sunt calculate a.î. vehiculul să urmeze într-un mod optimal traiectoria viitoare dorită. Studii în aplicarea controlului predictiv pentru vehicule autonome sunt deja în desfășurare atât în academie cât și în industrie, iar cititorii interesați pot face referire la [43, 192–198].

În acest capitol, vor fi prezentate metode de aplicare a controlului predictiv liniar și neliniar, atât pentru controlul dinamicii laterale, cât și a dinamicii longitudinale a vehiculelor autonome. Având disponibilă o traiectorie dorită, pe care un vehicul autonom trebuie să o urmărească, aceasta traiectorie poate fi translată în sistemul de referință al vehiculului generând astfel referințe pentru dinamicile laterale și longitudinale. Astfel, controlând într-un mod adecvat aceste dinamici, vehiculul poate fi ghidat pe traiectoria dorită. În acest scop, în acest capitol, vom considera controlul unui vehicul independent, fără a lua în considerare deplasarea acestuia într-un pluton.

Toți senzorii necesari (unghiul de virare al roților, poziția laterală, accelerația laterală, unghiul de girație, viteza de girație, vitezele roților) sunt considerați disponibili. De asemenea traiectoria dorită a vehiculului va fi considerată deja cunoscută.

II.2.6.1 Modelarea matematică a dinamicii mașinii

Pentru implementarea controlului predictiv atât liniar cât și neliniar, este nevoie de un model dinamic al vehiculului.

Uzual, dinamica vehiculelor este modelată folosind ”modelul bicicletă” (eng. ”bicycle model” sau ”single track model”). Acest model consideră roțile de pe axa din față a mașinii contopite într-o singură roată, aceeași simplificare fiind făcută și pentru roțile aflate pe axa din spate a vehiculului. De asemenea, centrul de greutate al vehiculului este considerat situat la nivelul solului, ignorand-u-se astfel influențele înclinării vehiculului asupra dinamicilor laterale și longitudinale. În Fig. II.29 este prezentat modelul bicicletă al unui vehicul.

Modelul matematic poate fi obținut folosind a doua lege a mișcării a lui Newton astfel:

$$\begin{aligned}m\ddot{x} &= m\dot{y}\dot{\psi} + 2F_{xf} + 2F_{xr}, \\m\ddot{y} &= -m\dot{x}\dot{\psi} + 2F_{yf} + 2F_{yr}, \\J\ddot{\psi} &= 2l_f F_{yf} - 2l_r F_{rf},\end{aligned}\tag{II.105}$$

unde x este poziția longitudinală a vehiculului, y este poziția laterală, ψ este unghiul de girație al vehiculului, F_{xf} și F_{xr} reprezintă forțele longitudinale generate în suprafața de contact al cauciucurilor roților din față cu suprafața de rulare, F_{yf} și F_{yr} reprezintă forțele laterale generate în suprafața de contact al cauciucurilor roților din față cu suprafața de rulare, l_f și l_r reprezintă distanța de la axa din față până la centrul de greutate și distanța

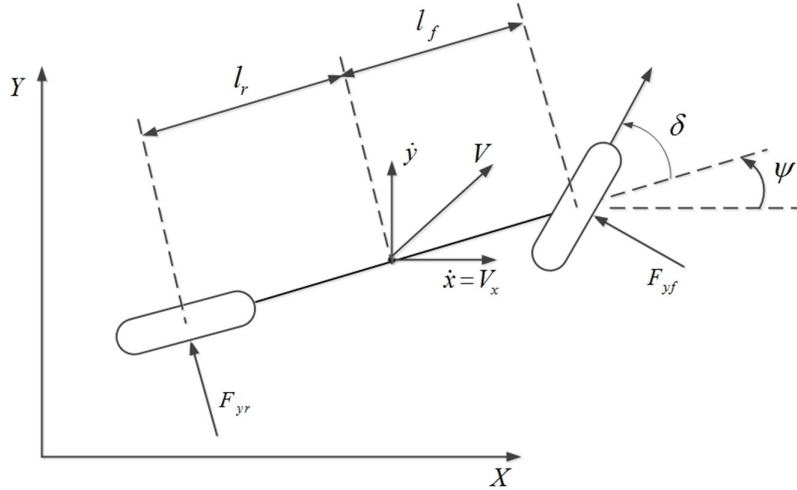


Figura II.29: Modelul bicicletei al dinamicii vehiculului.

de la axa din spate pana la centrul de greutate, m este masa vehiculului și J este inerția vehiculului.

Forțele cauciucurilor (F_{xf} , F_{xr} , F_{yf} și F_{yr}) sunt responsabile de generarea mișcării vehiculului. Aceste forțe pot fi modelate folosind modelul neliniar al lui Pacejka [51, 52], sub forma:

$$\begin{aligned} F_{xf} &= f_{xf}(\alpha_f, \kappa_f, F_{zf}), & F_{xr} &= f_{xr}(\alpha_r, \kappa_r, F_{zr}), \\ F_{yf} &= f_{yf}(\alpha_f, \kappa_f, F_{zf}), & F_{yr} &= f_{yr}(\alpha_r, \kappa_r, F_{zr}), \end{aligned} \quad (\text{II.106})$$

unde, α_f este unghiul de alunecare al roții din față, α_r este unghiul de alunecare al roții din spate, κ_f este alunecarea longitudinală a roții din față, κ_r este alunecarea longitudinală a roții din spate, iar F_{zf} și F_{zr} sunt forțele normale ce acționează supra axei din față, și axei din spate, respectiv.

Unghiurile de alunecare ale roților din (II.106), pot fi calculate astfel:

$$\tan \alpha_f = \frac{-\dot{x} \sin \delta + (\dot{y} + a\dot{\psi}) \cos \delta}{\dot{x} \cos \delta + (\dot{y} + a\dot{\psi}) \sin \delta}, \quad \tan \alpha_r = \frac{\dot{y} - b\dot{\psi}}{\dot{x}}, \quad (\text{II.107})$$

unde δ este unghiul de virare al roților din față.

Pentru a calcula alunecările longitudinale ale roților din față și din spate, luate în considerare în (II.106) trebuie diferențiat între accelerare și frânare astfel:

$$\kappa_{f,r} = -\frac{\dot{x} - R_e \omega_{f,r}}{R_e \omega_{f,r}}, \quad \kappa_{f,r} = -\frac{\dot{x} - R_e \omega_{f,r}}{\dot{x}}, \quad (\text{II.108})$$

unde $\omega_{f,r}$ reprezintă vitezele unghiulare ale roților din față și din spate, iar R_e reprezintă raza efectivă a roții.

Considerând că vehiculul se deplasează pe un drum cu o suprafață perfect plană, forțele normale F_{zf} și F_{zr} sunt date sub forma:

$$F_{zf} = \frac{mgb}{2(a+b)}, \quad F_{zr} = \frac{mga}{2(a+b)}. \quad (\text{II.109})$$

Folosind ecuațiile descrise de (II.105) până la (II.109), modelul matematic al dinamicii vehiculului poate fi scris compact sub forma unui model intrare-stare-ieșire neliniar:

$$\dot{z} = f(z, \delta, \omega_f, \omega_r), \quad (\text{II.110})$$

unde $z = [x, \dot{x}, y, \dot{y}, \psi, \dot{\psi}]^T$ reprezintă vectorul stărilor, iar $f : \mathbb{R}^6 \rightarrow \mathbb{R}^6$ este o funcție neliniară care descrie dinamica vehiculului.

Modelul neliniar (II.110), poate fi liniarizat în general cu scopul controlului unora dintre dinamicile laterale sau longitudinale, și studiul stabilității, controlabilității și observabilității acestora.

Un model liniar al dinamicii laterale, poate fi obținut prin liniarizarea forțelor laterale ale cauciucurilor roților din față și din spate:

$$F_{yf} = C_{yf}\alpha_f, \quad F_{yr} = C_{yr}\alpha_r, \quad (\text{II.111})$$

unde, C_{yf} și C_{yr} reprezintă coeficienții de rigiditate la viraje ale cauciucurilor. Forma forțelor laterale ale cauciucurilor descrisă de (II.111) este în general valabilă atunci când vehiculul se deplasează în condiții normale.

Unghiurile de alunecare α_f și α_r pot fi rescrise folosind aproximări ale unghiurilor mici astfel:

$$\alpha_f = \delta - \frac{\dot{y} + a\dot{\psi}}{\dot{x}}, \quad \alpha_r = \frac{\dot{y} - b\dot{\psi}}{\dot{x}}, \quad (\text{II.112})$$

Considerând că vehiculul se deplasează cu o viteză constantă, ecuațiile descrise de (1), pot fi reduse la forma care descrie dinamica laterală:

$$\begin{aligned} m\ddot{y} &= -mV_x\dot{\psi} + 2F_{yf} + 2F_{yr}, \\ J_v\ddot{\psi} &= 2aF_{yf} - 2bF_{yr}, \end{aligned} \quad (\text{II.113})$$

Astfel, folosind (II.111), (II.112) și (II.113) poate fi obținut modelul liniar intrare-stare-ieșire al dinamicii laterale al vehiculului:

$$\begin{aligned} \dot{z}(t) &= Az(t) + Bu(t) \\ y(t) &= Cz(t) \end{aligned} \quad (\text{II.114})$$

cu

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{2C_{yf}+2C_{yr}}{mV_x} & 0 & -V_x - \frac{2aC_{yf}-2bC_{yr}}{mV_x} \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{2aC_{yf}-2bC_{yr}}{JV_x} & 0 & -\frac{2a^2C_{yf}+2b^2C_{yr}}{JV_x} \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ \frac{2C_{yf}}{m} \\ 0 \\ \frac{2aC_{yf}}{J} \end{bmatrix}, \quad C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

unde $z = [y, \dot{y}, \psi, \dot{\psi}]^T$ este vectorul stărilor, $u = \delta$ este intrarea iar y este ieșirea modelului.

Modelele prezentate în acest capitol, pot fi utilizate atât în scopul simulării și studiului vehiculului, cât și în scopul implementării sistemului de control predictiv. În funcție de performanțe și necesități, controlul predictiv poate fi implementat fie folosind modelul liniar, fie folosind modelul neliniar.

Algoritmii de control predictiv prezentați în continuare, vor utiliza aceste modele ale vehiculului.

II.2.6.2 Controlul predictiv neliniar al dinamicii laterale a vehiculului

În continuare va fi descrisă proiectarea unui sistem de control predictiv neliniar pentru dinamica laterală a vehiculului, având ca scop implementarea unui sistem autonom de virare, care să ghideze vehiculul de-a lungul unei traiectorii dorite.

Metoda de control predictiv neliniar standard, poate fi formulată folosind un model dinamic descris de sistemul intrare-stare-ieșire neliniar discret de forma:

$$z_{k+1} = \phi(z_k, u_k) \quad k \in \mathbb{Z}_+, \quad (\text{II.115})$$

unde $z_k \in Z \subseteq \mathbb{R}^n$ este vectorul stărilor sistemului, $u_k \in U \subseteq \mathbb{R}^m$ este semnalul de comandă, iar $\phi : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$ este o funcție neliniară, care descrie dinamica sistemului.

Folosind modelul (II.115), regulatorul predictiv calculează semnalul de comandă, care minimizează funcția de cost (care poate fi atât liniară cât și neliniară)

$$J_N = \sum_{i=1}^N l(y_{k+i|k}, z_{k+i|k}, u_{k+i|k}), \quad (\text{II.116})$$

fiind supusă restricțiilor (care la rândul lor pot fi atât liniare cât și neliniare)

$$\begin{aligned} c_{eq}(y_{k+i|k}, z_{k+i|k}, u_{k+i|k}) &= 0, \\ c_{ineq}(y_{k+i|k}, z_{k+i|k}, u_{k+i|k}) &\geq 0, \end{aligned} \quad (\text{II.117})$$

Algoritmul de control al virării automate a vehiculului, folosește modelul intrare-stare-ieșire neliniar al vehiculului descris de (II.110), iar funcția de cost (II.116) și restricțiile (II.117) este aleasă de forma:

$$J_N = \sum_{i=1}^N [w_{k+i|k} - y_{k+i|k}]^2 \quad (\text{II.118})$$

cu

$$\begin{aligned} u^{\min} &\leq u_{k+i-1} \leq u^{\max}, \\ -u^\Delta &\leq \Delta u_{k+i-1} \leq u^\Delta, \\ x^{\min} &\leq x_{k+i} \leq x^{\max}, \end{aligned} \quad (\text{II.119})$$

unde $w_{k+i|k}$ este poziția laterală de referință viitoare.

Semnalul de comandă este considerat unghiul de virare al roților din față δ , în timp ce marimea reglată este poziția laterală a vehiculului y .

Algoritmul de control predictiv neliniar propus, arată rezultate foarte bune în simulare. Validarea algoritmului propus a fost făcută prin simulări a unei manevre de schimbare a benzii de rulare, apoi revenire pe banda inițială, considerând orizonturi de predicție diferite și viteza longitudinală constantă. În Fig. II.30 este prezentată o dublă-schimbare de bandă, la viteza longitudinală $V_x = 30$ [km/h] și orizontul de predicție $N = 10$.

În același mod, simulări cu orizontul de predicție $N = 3$ și $N = 20$ sunt prezentate în Fig. II.31 și Fig. II.32.

În urma acestui studiu s-a putut observa că un vehicul autonom, a cărui sistem de control este un sistem de control predictiv neliniar, poate urmări cu înaltă precizie traiectoria dorită.

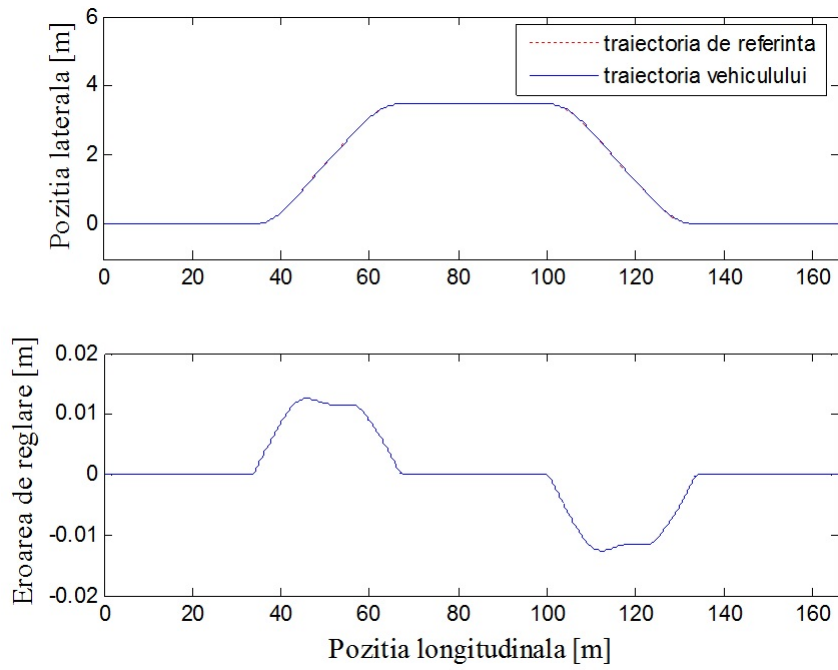


Figura II.30: Dublă-schimbare de bandă cu orizont de predicție $N = 10$.

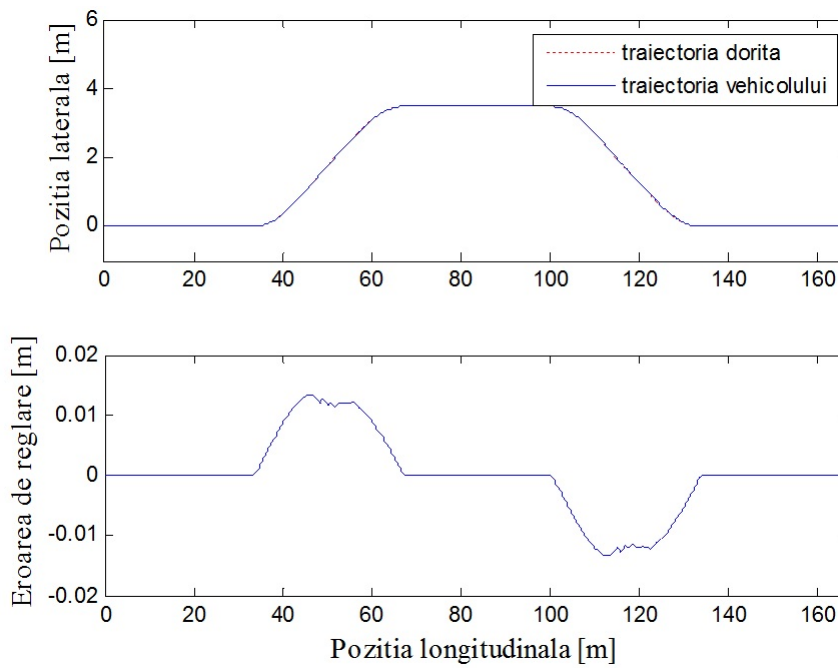


Figura II.31: Dublă-schimbare de bandă cu orizont de predicție $N = 3$.

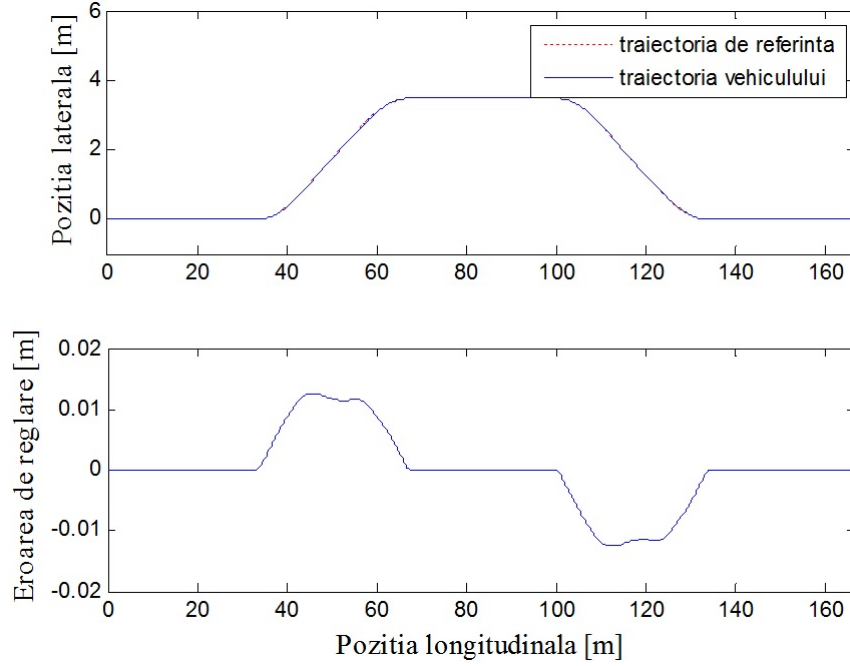


Figura II.32: Dublă-schimbare de bandă cu orizont de predicție $N = 20$.

De asemenea, în Fig. II.31 se pot observa mici oscilații, ceea ce este explicabil prin faptul că regulatorul predictiv este considerat cu orizontul de predicție foarte mic $N = 3$.

În Fig. II.33 este prezentată o comparație între cele trei simulări, pentru a vedea efectul variației orizontului de predicție. După cum este de așteptat, regulatorul predictiv neliniar având orizontul de predicție $N = 20$ prezintă cele mai bune performanțe, în timp ce regulatorul predictiv cu orizontul de predicție $N = 3$ prezintă cele mai scăzute performanțe, având oscilații.

Cu toate că regulatorul predictiv neliniar propus arată performanțe foarte bune în aplicații pentru urmărirea automată a traiectoriei, stabilitatea sistemului nu este încă asigurată. În mod usual, pentru a asigura stabilitatea sistemelor de control cu regulator predictiv neliniar, se includ funcții de stabilitate Lyapunov sub forma de restricții terminale (în funcția de cost (II.116)) sau restricții neliniare adiționale celor din (II.117). În capitolul următor, va fi prezentată strategia de control predictiv neliniar incluzând funcțiile de stabilitate Lyapunov.

II.2.6.3 Controlul predictiv neliniar al dinamicii laterale a vehiculelor cu funcții de stabilitate Lyapunov

În acest capitol va fi prezentată o metodă de a include funcții de stabilitate Lyapunov în controller-ul predictiv neliniar prezentat anterior.

Fie sistemul discret neliniar, descris prin reprezentarea intrare-stare-ieșire

$$z_{k+1} = \phi(z_k, u_k) \quad k \in \mathbb{Z}_+, \quad (\text{II.120})$$

unde $z_k \in Z \subseteq \mathbb{R}^n$ este vectorul stărilor sistemului, $u_k \in \mathbb{U} \subseteq \mathbb{R}^n$ este semnalul de

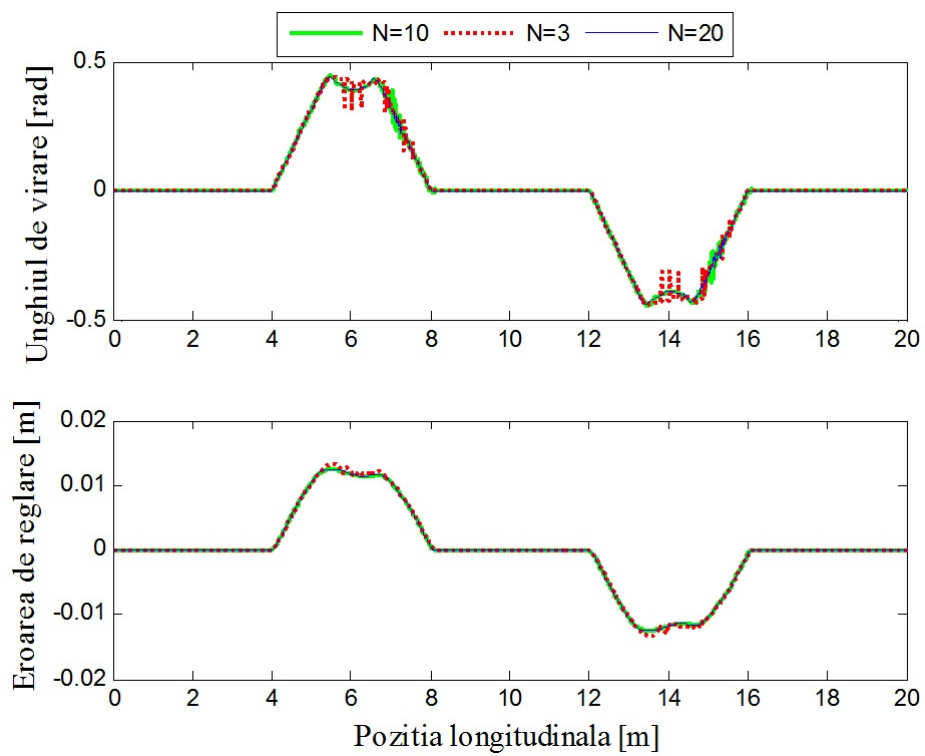


Figura II.33: Comparație între performanțele regulatorului predictiv cu orizonturile de predicție $N = 3$, $N = 10$ și $N = 20$.

comandă, iar $\phi : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$ este o funcție neliniară, care descrie dinamica sistemului. Fie $\alpha_1, \alpha_2 \in \mathcal{K}_\infty$ și $\rho \in \mathbb{R}_{[0,1]}$.

Definiția II.2.1 O funcție $V : \mathbb{R}^n \rightarrow \mathbb{R}_+$ care satisface

$$\alpha_1(\|z_k\|) \leq V(z_k) \leq \alpha_2(\|z_k\|), \quad \forall z_k \in \mathbb{R}^n \quad (\text{II.121})$$

și pentru care există o lege de comandă $\pi : \mathbb{R}^n \rightrightarrows \mathbb{U}$, a.î.

$$V(z_{k+1}) \leq \rho V(z_k), \quad \forall z_k \in Z, \forall u_k \in \pi(z_k), \quad (\text{II.122})$$

se numeste o funcție Lyapunov, pentru sistemul (II.120), în buclă închisă cu $u_k \in \pi(x_k)$, $k \in \mathbb{Z}_+$.

Pentru a include o funcție de stabilitate Lyapunov în sistemul de control predictiv, trebuie ca această funcție să fie găsită a priori, sarcină care se poate dovedi dificilă în majoritatea cazurilor, datorită faptului că aceste funcții sunt dificil de găsit.

Presupunând cunoscută funcția de stabilitate Lyapunov, sarcina regulatorului predictiv proiectat devine ca la orice moment de timp $k \in \mathbb{Z}_+$ să calculeze comanda u_k , a.î.

$$\begin{aligned} u &\in, z \in Z, \\ V(z^+) &\leq \rho V(z). \end{aligned} \quad (\text{II.123})$$

Propoziția II.2.2 Fie o funcție de stabilitate Lyapunov $V(\cdot)$ dată pentru sistemul (II.120). Presupunând că problema descrisă de (II.123) este fezabilă pentru orice $z \in Z$, atunci sistemul în buclă închisă

$$z_{k+1} \in \phi_{cl}(z_k, \pi(z_k)), \quad k \in \mathbb{Z}_+, \quad (\text{II.124})$$

este asimptotic stabil.

În concluzie, stabilitatea regulatorului predictiv neliniar poate fi asigurată prin includerea unei funcții Lyapunov descrisă anterior, în restricțiile aplicate funcției de cost (II.119).

Astfel, în regulatorul predictiv neliniar propus, a fost inclusă funcția de stabilitate Lyapunov de forma

$$V(z) = \|Pz\|_\infty, \quad (\text{II.125})$$

unde $P \in \mathbb{R}^{p \times n}$ este o matrice de rang complet determinată a priori.

Funcția de cost care trebuie minimizată a fost extinsă la forma:

$$J(z_k, u_k) = \sum_{i=1}^N (\|Pz(z_{k+i} - z_k^{ss})\|_\infty + \|R(u_{k+i-1} - u_k^{ss})\|_\infty), \quad (\text{II.126})$$

unde $P_z = I_4$ iar $R = 0.5$. De asemenea restricțiile asupra comenzii au fost considerate ca fiind $|\delta| \leq \frac{\pi}{2}$.

Modelul dinamic al vehiculului folosit în regulatorul predictiv neliniar este dat de (II.105), unde viteza longitudinală a fost considerată constantă.

Algoritmul predictiv astfel obținut a fost validat prin simulare, folosind aceeași manevră prezentată anterior - dublă schimbare de bandă, considerând diferite viteze longitudinale.

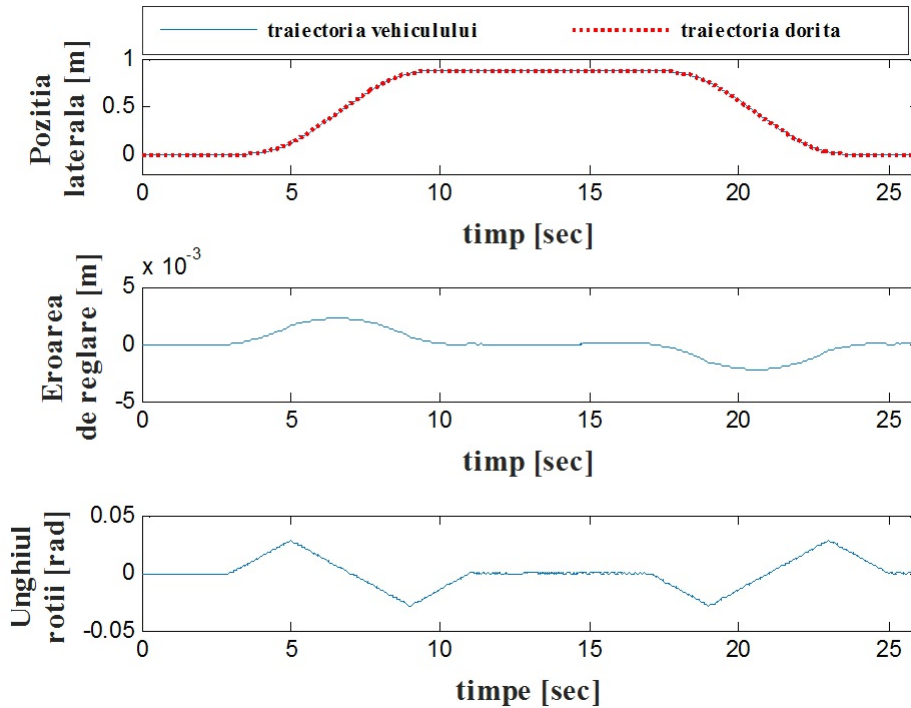


Figura II.34: Dublă-schimbare de bandă cu viteza $V_x = 11.16$ [km/h].

În Fig. II.34 o dublă schimbare de bandă poate fi observată, vehiculul având viteza longitudinală constantă $V_x = 11.16$ [km/h].

În Fig. II.35 și Fig. II.36 este executată aceeași manevră, considerând că vehiculul se deplasează cu viteza $V_x = 17.5$ [km/h] și respectiv $V_x = 23.3$ [km/h]. Se poate observa că performanțele algoritmului de control predictiv incluzând funcția de stabilitate Lyapunov oferă performanțe ridicate în cotrolul traiectoriei vehiculului.

Dificultățile în dezvoltarea și implementarea unui algoritm de control adecvat pentru vehiculele autonome provin din faptul că acest algoritm de control trebuie să obțină performanțe ridicate în controlul atât lateral, cât și longitudinal al vehiculului, în același timp oferind confort pasagerilor și în principal siguranța. În general, controlul lateral și longitudinal al vehiculelor sunt implementate decuplat, prin bucle de control separate. Pentru a implementa un algoritm de control care să reacționeze similar cu omul, dinamicile laterale și longitudinale trebuie tratate centralizat, aparând astfel necesitatea extinderii sistemului SISO de reglare la un sistem MIMO.

În capitolele următoare, vor fi prezentați algoritmi de control predictiv dezvoltați cu scopul de a controla atât dinamica laterală cât și dinamica longitudinală a vehiculului.

II.2.6.4 Controlul predictiv neliniar multivariat al dinamicii vehiculelor

Pentru implementarea controlului centralizat al dinamicii laterale și a celei longitudinale a unui vehicul autonom, apare necesitatea implementării unei strategii de control multivari-

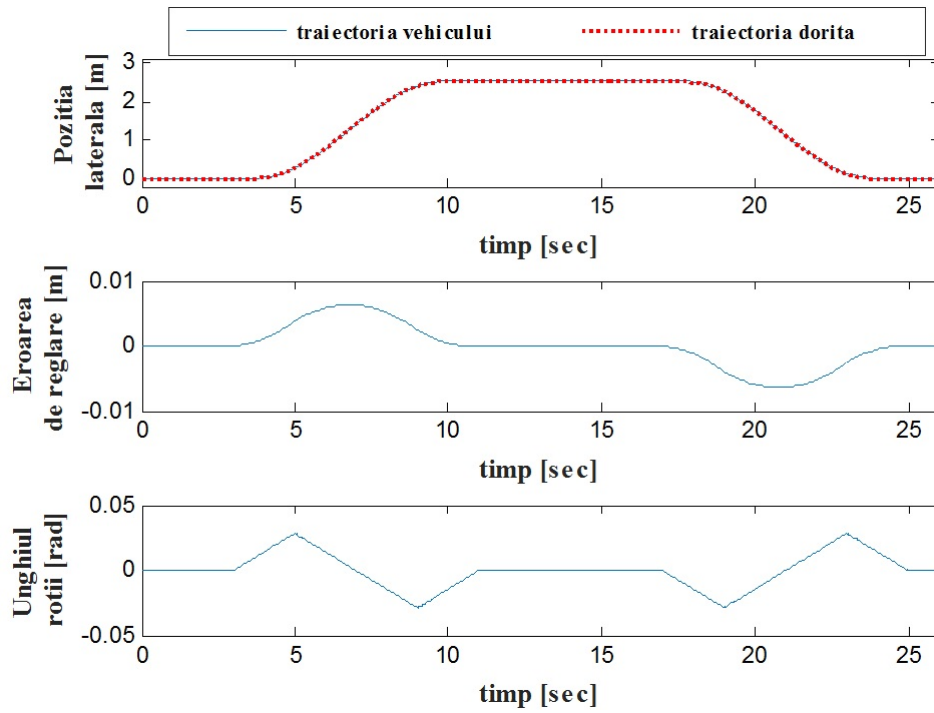


Figura II.35: Dublă-schimbare de bandă cu viteza $V_x = 17.5$ [km/h].

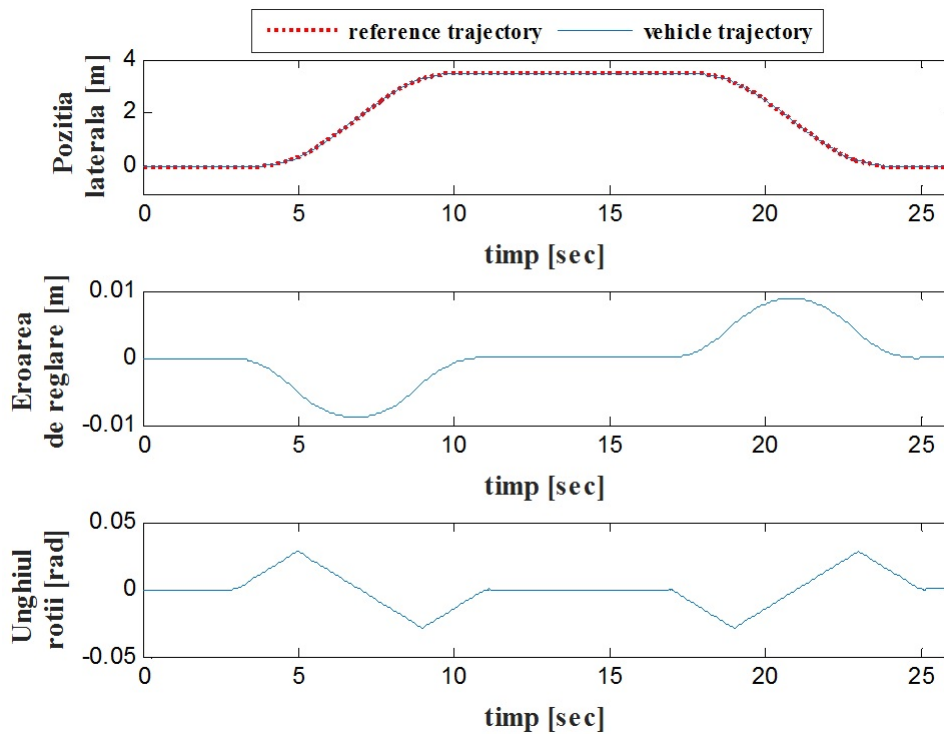


Figura II.36: Dublă-schimbare de bandă cu viteza $V_x = 22.3$ [km/h].

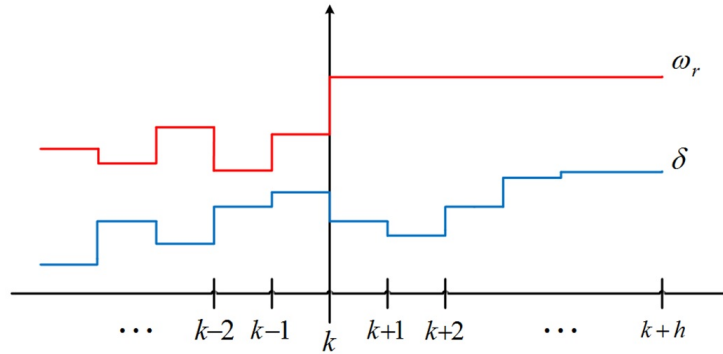


Figura II.37: Structurarea semnalelor de comandă.

abile. Cu toate că strategia de control predictiv poate fi extinsă într-o manieră naturală de la sistemele SISO la sistemele MIMO, cuplajul dintre cele două dinamici generează dificultăți în căutarea comenzii optimale, ceea ce poate conduce la performanțe scăzute sau chiar instabilitate. Cuplajul dintre dinamica laterală și longitudinală a unui vehicul poate fi observat în modelul neliniar (II.105).

Diverse metode de a realiza această decuplare pot fi vazute în [199,200].

Acest capitol propune implementarea unui algoritm de control predictiv neliniar multi-variabil, care realizează implicit decuplarea dinamicilor laterale și longitudinale, folosind orizontul comenzii.

Asemenea capitolelor precedente, modelul dinamic al vehiculului este considerat cel descris în (II.105), cu (II.106)-(II.109). Acest model poate fi scris sub forma intrare-stare-ieșire

$$\dot{z} = \phi(z, \delta, \omega_f, \omega_r), \quad (\text{II.127})$$

unde $z = [x, \dot{x}, y, \dot{y}, \psi, \dot{\psi}, X, Y]^T$ reprezintă vectorul stărilor, δ este unghiul de virare al roților din față, ω_f și ω_r sunt vitezele unghiulare ale roților din față și din spate, respectiv. Intrările în sistemul propus sunt δ care controlează dinamica laterală, ω_f și ω_r care controlează dinamica longitudinală. Pentru simplitate, este considerat că $\omega_f = \omega_r$.

Algoritmul de control predictiv neliniar multivariabil propus, structurează comenzile δ și ω_f (sau ω_r) prin intermediul orizontului comenzii ω_r , notat cu N_{ω} . Astfel cele două comenzi pot fi structurate așa cum este prezentat în Fig. II.37.

Prin această structurare, semnalul de control ω_r este forțat să rămână constant de-a lungul orizontului de predicție, a.î. doar semnalul de control δ poate varia pe toată lungimea orizontului.

Aplicând această strategie de control predictiv neliniar, cu un orizont de predicție $N = 10$, rezultatele obținute prin simulare arată performanțe foarte bune în urmărirea referințelor de viteză longitudinală, respectiv poziție laterală.

În Fig. II.38 este prezentată o manevră de dublă schimbare a benzii, unde traiectoria de referință este generată pentru o viteză constantă de $V_x = 22.3$ [km/h]. Viteza unghiulară inițială a roților este considerată $\omega_r = 19.9$ [rad/sec], ceea ce este echivalent cu o viteză a mașinii $V_x = 22.3$ [km/h]. Se poate vedea că traiectoria este urmarită foarte îndeaproape.

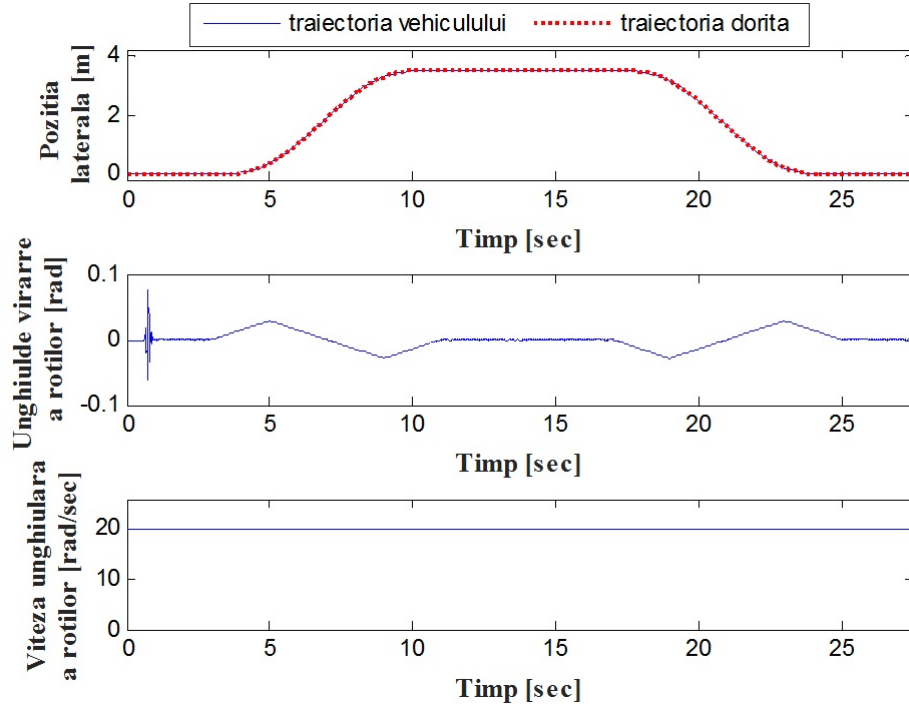


Figura II.38: Dublă schimbare a benzii, cu referință de viteză longitudinală $V_x = 22.3$ [km/h].

II.2.6.5 Controlul predictiv centralizat al dinamicii vehiculelor autonome folosind modele liniare

Un algoritm alternativ de control predictiv centralizat al dinamicii laterale și longitudinale a vehiculelor autonome poate fi obținut folosind modele liniare ale dinamicii laterale și un model linear al lanțului de transmisie al vehiculului. Astfel, cuplajul dintre cele două dinamici (longitudinale și laterale) poate fi luat în mod direct în calcul de către algoritmul predictiv.

Fie modelul linear al dinamicii laterale, scris sub forma intrare-stare-ieșire ca în (II.114):

$$\begin{aligned} \dot{z}^{veh}(t) &= A_{veh}^c z^{veh}(t) + B_{veh}^c u^\delta(t) \\ y^{veh}(t) &= C_{veh}^c z^{veh}(t) \end{aligned} \quad (II.128)$$

cu

$$A_{veh}^c = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{2C_{yf}+2C_{yr}}{mV_x} & 0 & -V_x - \frac{2aC_{yf}-2bC_{yr}}{mV_x} \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{2aC_{yf}-2bC_{yr}}{J_v V_x} & 0 & -\frac{2a^2C_{yf}+2b^2C_{yr}}{J_v V_x} \end{bmatrix},$$

$$B_{veh}^c = \begin{bmatrix} 0 \\ \frac{2C_{yf}}{m} \\ 0 \\ \frac{2aC_{yf}}{J_v} \end{bmatrix}, \quad C_{veh}^c = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix},$$

unde $z^{veh} = [y, \dot{y}, \psi, \dot{\psi}]^T$ reprezintă vectorul stărilor, iar $u^\delta = \delta$ reprezintă unghiul de virare al roților din față.

Fie, de asemenea, un model liniar intrare-stare-ieșire al lanțului de transmisie al vehiculului considerat:

$$\begin{aligned} \dot{z}^{drv}(t) &= A_{drv}^c z^{drv}(t) + B_{drv}^c u^\tau(t), \\ y^{drv}(t) &= C_{drv}^c z^{drv}(t) \end{aligned} \quad (\text{II.129})$$

cu

$$A_{drv}^c = \begin{bmatrix} -\frac{b_e}{J_{eg}} - \frac{b_f}{i_{tot}^2 J_{eg}} & \frac{b_f}{i_{tot} J_{eg}} & -\frac{k_f}{i_{tot} J_{eg}} \\ \frac{b_f}{i_{tot} J_v} & -\frac{b_a + b_f}{J_v} & \frac{k_f}{J_v} \\ \frac{1}{i_{tot}} & -1 & 0 \end{bmatrix}, B_{drv}^c = \begin{bmatrix} \frac{1}{J_{eg}} \\ 0 \\ 0 \end{bmatrix}, C_{drv}^c = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix},$$

unde $z^{drv} = [\omega_e, \omega_f, \theta_e/i_{tot} - \theta_w]^T$ reprezintă vectorul stărilor, iar $u^\tau = T_e$ este cuplul motorului termic.

Descrierea modelării lanțului de transmisie al vehiculului nu va fi prezentată în acest capitol, iar cititorii interesați pot face referire la [18].

Având modelele (II.128) și (II.129), algoritmul predictiv propus poate fi descris astfel:

Algoritmul II.2.3 La fiecare pas $k \in \mathbb{Z}_+$

1. măsoară stările $z_k = [z_k^{veh}, z_k^{drv}]^T$;
2. rezolvă problema de optimizare și obține $u^*(z_k) = [u_k^{veh*}, u_k^{drv*}]^T$;
 - 2.1. considerând $u_{k+i|k}^\tau, k = 1, \dots, N$ se pot obține stările $z_{k+i|k}^{drv}, k = 1, \dots, N$;
 - 2.2. folosind $z_{k+i|k}^{drv}, k = 1, \dots, N$ se poate obține un șir de matrici (modele) de forma (II.128), și considerând $u_{k+i|k}^\delta, k = 1, \dots, N$ ca intrări în acest șir de modele, se pot obține stările vehiculului $z_{k+i|k}^{veh}, k = 1, \dots, N$;
 - 2.3. stările obținute $z_{k+i|k} = [z_{k+i|k}^{drv}, z_{k+i|k}^{veh}]^T, k = 1, \dots, N$ sunt folosite pentru a rezolva problema de optimizare;
 - 2.4. rezolvă problema de optimizare și obține $u_{k+i|k}^*(z_{k+i|k}), k = 1, \dots, N$;
3. aplică semnalul de control $u_k = u_{k+1}^*$;
4. revino la pasul 1.

Analiza algoritmului predictiv propus a fost făcută prin simulare, unde a fost considerat un orizont de predicție $N = 10$. Funcția de cost considerată este:

$$J = \sum_{i=1}^N \left\{ \left[z_{k+i|k}^{drv,ref} - z_{k+i|k}^{drv} \right]^2 + \left[z_{k+i|k}^{veh,ref} - z_{k+i|k}^{veh} \right]^2 \right\} \quad (\text{II.130})$$

cu restricțiile $0 \leq u_{k+i|k}^\tau \leq 120$ și $\left| u_{k+i|k}^\delta \right| \leq \frac{\pi}{2}, k = 1, \dots, N$.

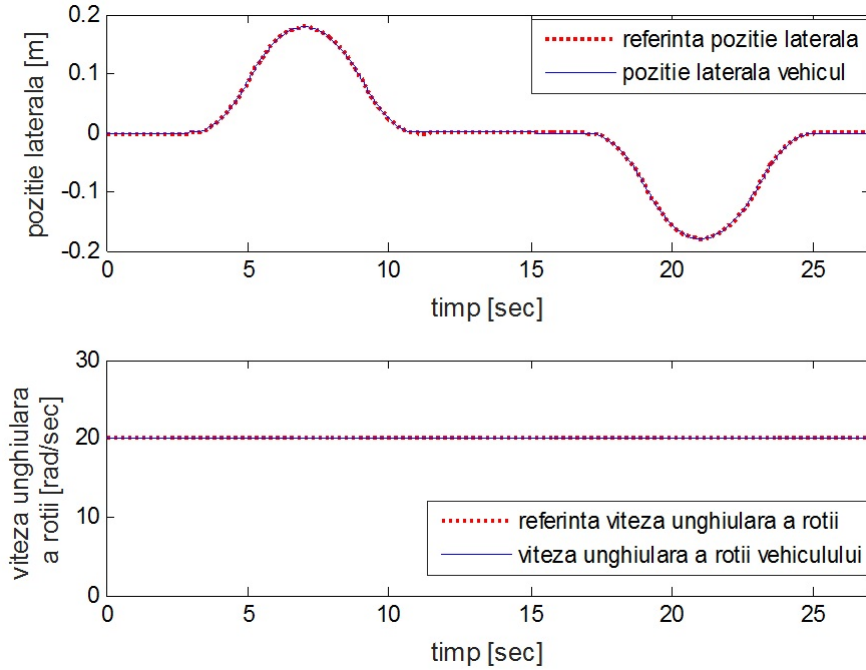


Figura II.39: Dublă schimbare a benzii cu referința de viteză unghiulară a roții $z_{k+i|k}^{drv,ref} = 20$ [rad/sec].

În Fig. II.39, este prezentată o manevră de dublă schimbare a benzii, unde referința de viteză unghiulară a roților este constantă $z_{k+i|k}^{drv,ref} = 20$ [rad/sec], de asemenea viteza unghiulară inițială a roților este $z_0^{drv} = 20$ [rad/sec]. În acest studiu, mărimile sunt considerate în reperul cartezian asociat vehiculului.

În Fig. II.40, este prezentată o manevră de dublă schimbare a benzii, unde referința de viteză unghiulară a roților se schimbă în formă de semnal treaptă. De asemenea, viteza inițială a roților este considerată diferită de cea de referință, respectiv $z_0^{drv} = 40$ [rad/sec]. Aici, se pot observa ușoare oscilații în momentul în care viteza unghiulară se schimbă.

În Fig. II.41, aceeași manevră de dublă schimbare a benzii este simulată, de această dată considerând că referința de viteză unghiulară a roții se schimbă în formă de semnal treaptă în timpul manevrei de virare.

II.2.7 Planificarea optimală a traiectoriilor pentru vehicule autonome

Problema generării traiectoriilor corect adaptate la trafic, vine din prisma necesității ca autovehiculele să poată circula în condiții de siguranță, chiar și în condiții foarte dinamice de trafic, cum ar fi de exemplu pe autostrăzi.

Planificarea de traiectorii este responsabilă cu generarea traiectoriilor dorite pe care vehiculul autonom trebuie să le urmărească. Algoritmii existenți de planificare provin în general din domeniul roboticii, și majoritatea algoritmilor au fost dezvoltați pentru roboți non-holonomici, asemănători unui autovehicul.

În contextul planificării de traiectorii, timpul execuției controlului este considerat în mod explicit, ceea ce permite modelarea în mod direct a dinamicii vehiculelor și obstacolelor.

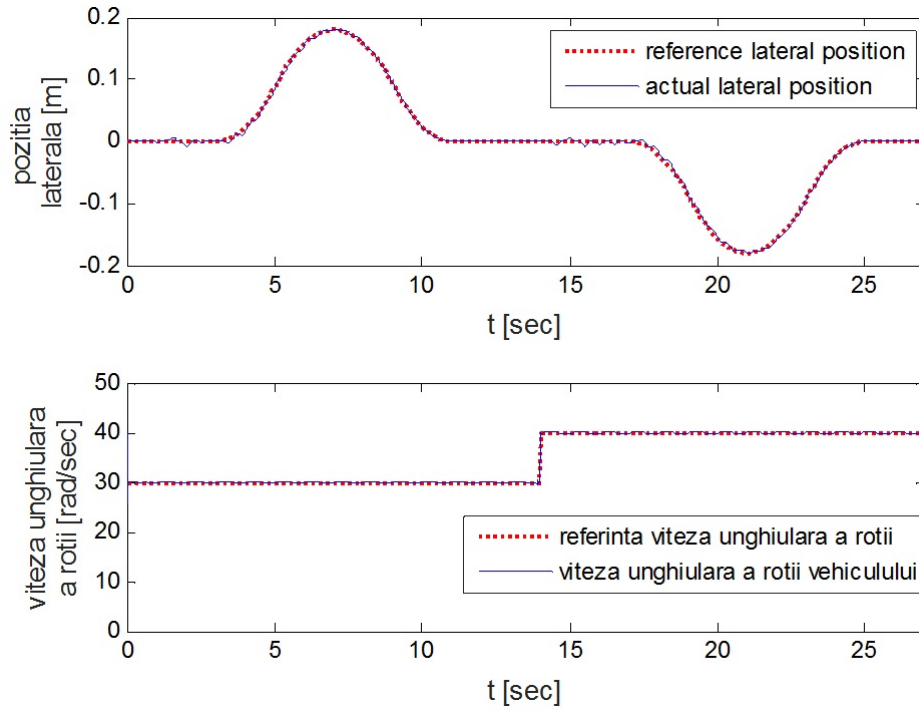


Figura II.40: Dublă schimbare a benzii cu referința de viteză unghiulară a roții variabilă.

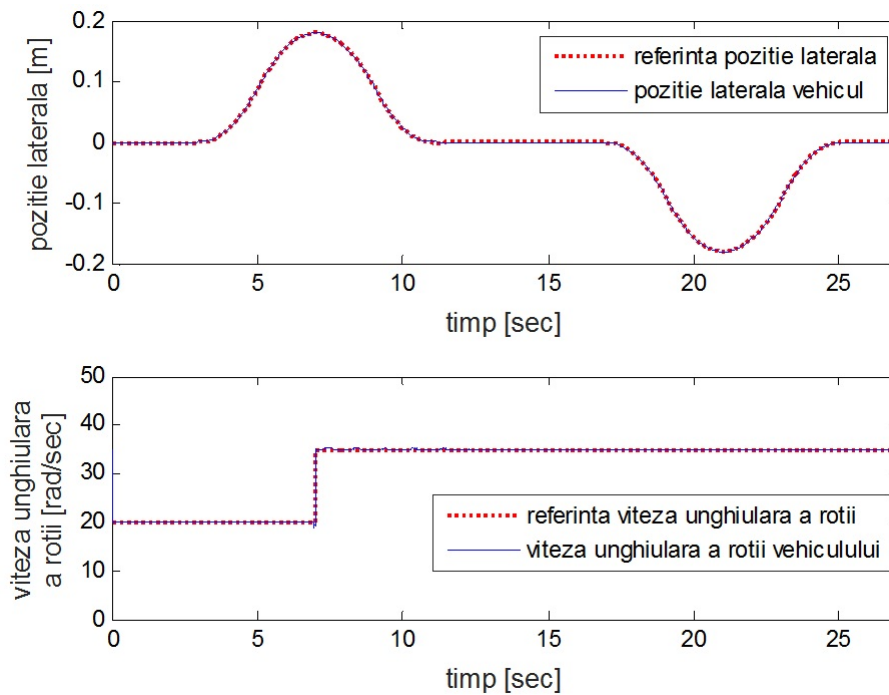


Figura II.41: Dublă schimbare a benzii cu referința de viteză unghiulară a roții variabilă simultan.

În acest caz, traiectoria obținută este descrisă de o funcție de timp, scrisă generic sub forma $T(t) : [0, T] \rightarrow X$, unde T este orizontul de timp al planificării.

În ultimii 30 de ani, au fost derulate cercetări ambițioase în ceea ce privește dezvoltarea vehiculelor autonome, iar cum acestea evoluează către capacitatea de a circula pe drumurile publice, trebuie să ia în considerare toți participanții la trafic în mod explicit, incluzând manevrele uzuale, cum ar fi intrarea în coloană, schimbarea benzii și evitarea obstacolelor. În condiții simplificate și controlate, așa cum ar fi concursul DARPA Urban Challenge 2007, generarea de traiectorii pentru vehiculele autonome poate fi făcută relativ ușor [201]. Însă, aceste metode simplificate nu mai pot fi aplicate în condiții realiste de trafic intens, sau la viteze de rulare ridicate [202]. De aici, apare necesitatea de a implementa metode pentru a planifica traiectoriile dorite, ținând cont în mod explicit de momentul de timp deja din momentul planificării.

Câteva metode pentru planificarea traiectoriilor au fost propuse până în prezent [203–206], care pot găsi o traiectorie globală ce conectează o stare de start și o stare de final (o stare țintă). Cu toate acestea, aceste metode nu reușesc să țină cont de imprezibilitatea inerentă a condițiilor de trafic, bazându-se pe predicții foarte precise a mișcării celorlalți participanți la trafic, ceea ce este foarte greu dacă nu chiar imposibil de realizat, pe perioade mai lungi de timp. Alte metode se bazează pe găsirea unei traiectorii optime, dintr-un set de traiectorii posibile, folosind algoritmi de optimizare [207–209]. Aceste metode generează un set finit de traiectorii posibile, pe care vehiculul autonom le poate urmări. Din acest set, folosind un algoritm de optimizare, este selectată traiectoria optimă, prin minimizarea unei funcții de cost. Traiectoriile sunt generate prin integrarea numerică a modelelor matematice ale vehiculelor (în mod uzual). Setul de traiectorii este generat folosind polinoame, clotoide, etc. Cu toate că aceste metode pot genera traiectorii mai eficiente și mai rapide decât cele menționate anterior, traiectoriile generate pot fi sub-optimale datorită spațiului redus de traiectorii posibile, poate apărea instabilitatea sistemului sau erori de regim staționar în curbe.

În Tabelul de mai jos, este făcută o scurtă trecere în revistă a unor algoritmi de planificare de traiectorii cunoscuți, unde sunt prezentate metoda geometrică de descriere a traiectoriei, modelul dinamic al vehiculului folosit, și limitările algoritmului respectiv.

Pe lângă metodele amintite mai sus, cele mai promițătoare metode de planificare de traiectorii pentru vehiculele autonome, sunt considerate metodele de tipul eșantionare în spațiul stărilor (eng. State Space Sampling). Aceste metode sunt suficient de rapide, fiind capabile să genereze traiectorii cu obiectiv final îndelungat (traiectorii mai lungi), și în același timp, răspunzând în mod reactiv la condițiile de trafic și de evitare a obstacolelor. Una dintre aceste metode este propusă în [221] și va fi descrisă în cele ce urmează.

Planificarea și generarea de traiectorii este separată de algoritmul de control care stabilizează vehiculul pe traiectoria dorită. Aplicarea teoriei de control optimal nu este nouă pentru metodele de planificare și generare de traiectorii, dar metoda descrisă în [221] nu propune minimizarea online a unei funcții de cost, folosind algoritmi de minimizare, ci se concentrează pe generarea tuturor traiectoriilor astfel încât toată calea urmarită de vehicul să fie optimală, și nu doar porțiunile generate la fiecare interval de timp. Astfel, algoritmul încearcă, odată ce a găsit o traiectorie optimă, aceasta să fie reținută, urmărind astfel principiul optimalității al lui Bellman. Aceasta înseamnă, din punctul de vedere al vehiculului,

Table 6: Trecere în revistă a unor algoritmi de planificare

| Lucrare | Modelul traiectoriei | Modelul vehiculului | Limitări |
|---------|----------------------------------|--|--|
| [210] | Polinoame de ordinul 2 | Poziția vehiculului în două dimensiuni | Nu sunt considerate obstacole. |
| [211] | Polinoame de ordinul 4 | Modelul bicicletă | Vehiculele sunt considerate ca au viteză constantă. |
| [212] | Curbe cubice Bezier | Dreptunghi | Costisitor din punct de vedere numeric. |
| [213] | Curbe cubice Bezier | Nu este dat | Nu ține cont de potențialele coliziuni. |
| [214] | Curbe cubice Bezier de ordinul 4 | Dreptunghi | Obiectele dinamice nu sunt luate în considerare. |
| [215] | Polinoame de ordin 3 și 5 | Modelul bicicletă | Doar schimbarea benzii este considerată. |
| [216] | Spline Akima | Nu este dat | Nu sunt date rezultate extinse. |
| [217] | Spline | Nu este dat | Nu se ia în considerare traficul. |
| [218] | Coridor de condus | Nu este dat | Costisitor din punct de vedere numeric. |
| [219] | MPC | Modelul bicicletă | Optimizarea este sensibilă la numărul de restricții variabile. |
| [220] | MPC | Poziția vehiculului în două dimensiuni | Optimizarea este sensibilă la numărul de restricții variabile. Consideră doar obstacole statice. |
| [221] | Polinoame de ordin 5 | Model punct cu masă | Doar rezultate de simulare. |
| [222] | Polinoame de ordin 6 și 7 | Modelul bicicletă | Doar rezultate de simulare. |

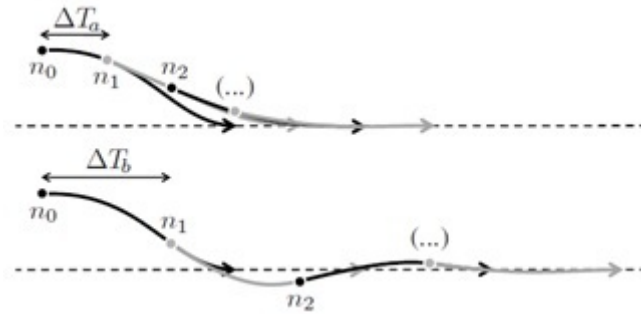


Figura II.42: Comportări tranzitorii diferite ale aceleiași strategii de planificare.

că va urmări segmentele rămase de parcurs ale traiectoriilor calculate în ciclul de planificare anterior, astfel asigurându-se consistența temporală. Acest aspect este foarte important pentru ca înconsistența temporală între traiectorii consecutive poate duce la suprareglări ale dinamicii vehiculelor sau chiar instabilitate. Acest lucru este arătat în Fig. II.42.

În timp ce scopul algoritmului este în principal de a urmări principiul de optimalitate al lui Bellman, este important ca traiectoriile generate să păstreze caracteristicile dorite adaptate la trafic, astfel încât vehiculul care urmărește aceste traiectorii să poată conduce corect și realist. Astfel, un vehicul autonom trebuie să fie controlat într-un mod ideal pe traseul dorit. De exemplu: să presupunem că vehiculul are o anumită deplasare laterală față de banda dorită, datorită unei manevre de schimbare a benzii abia inițiată. Vehiculul trebuie apoi să revină la banda dorită, în limitările fizice, făcând astfel cel mai bun compromis între confortul pasagerilor și timpul în care vehiculul revine complet pe banda dorită. În același timp, și dinamica laterală trebuie luată în considerare în aceeași manieră. O formulare matematică a confortului poate fi făcută sub forma de jerk, care este definit ca rata de schimbare (derivata în timp) a accelerației laterale și longitudinale, unde intervalul de timp este $T = t_f - t_i$ durata de timp în care vehiculul trebuie să parcurgă traiectoria planificată, plecând de la punctul de start la punctul de sfârșit al traiectoriei respective. Folosind o metoda iterativă de minimizare, problema poate deveni foarte complexă și costisitoare computațional, în momentul în care apar restricții dinamice și de trafic. De aceea, metoda propusă generează traiectoriile *liber* (fara restricții), apoi căutând traiectoriile care nu încalcă aceste restricții. Aceasta înseamnă că odată găsită o traiectorie optimală, consistența temporală va fi îndeplinită.

O metodă cunoscută în problemele de urmărire de traiectorii este metoda care folosește Coordonatele Frenet, care garantează performanțe invariante de urmărire în reperul cu două dimensiuni $SE(2) := SO(2) \times \mathbb{R}^2$. Metoda propusă pentru planificarea traiectoriilor face uz de reperul Frenet, pentru a putea combina funcții de cost diferite pentru mișcarea laterală și cea longitudinală, pentru diferite manevre și în același timp să mimeze conducerea umană a vehiculului. După cum este prezentat în Fig. II.43, reperul considerat este definit prin vectorii tangențial și normal \vec{t}_r și \vec{n}_r la un anumit punct într-o curbă, numită linie de centru sau traiectorie de referință. Această referință, în cel mai simplu caz reprezintă centrul benzii pe care vehiculul autonom trebuie să circule. Astfel, algoritmul nu generează traiectorii direct în reperul Cartezian, ci în reperul dinamic Frenet, încercând să obțină funcții

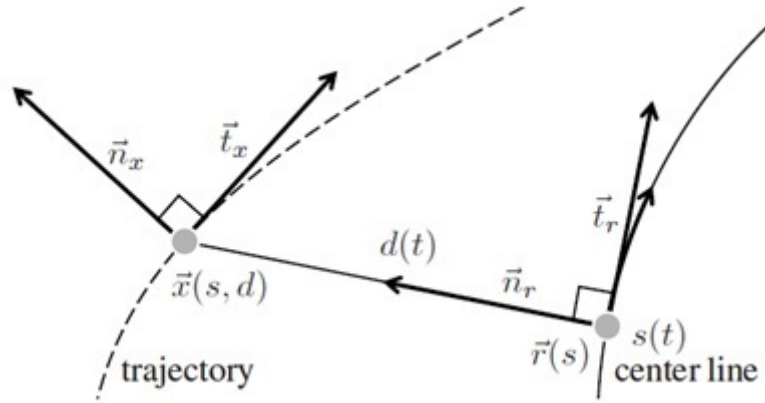


Figura II.43: Generarea de traiectorii în reperul Frenet.

unidimensionale pentru punctul \vec{r} și deplasarea d , folosind relația

$$\vec{x}(s(t), d(t)) = \vec{r}(s(t)) + d(t)\vec{n}_r(s(t)), \quad (\text{II.131})$$

unde $s(t)$ reprezintă lungimea de arc deja parcursă. Deoarece vectorii \vec{t}_r și \vec{n}_r , \vec{t}_x și \vec{n}_x aproape se aliniază la viteze mari, funcția de cost care minimizează rata de schimbare a accelerației laterale și longitudinale este definită pe baza \ddot{d} și \ddot{s} . Din [223] se știe că polinoamele de ordinul cinci sunt funcții optimale din punct de vedere al jerk-ului, astfel încât o curbă care conectează un punct de start $P_0 = [p_0, \dot{p}_0, \ddot{p}_0]$ și un punct de stop $P_1 = [p_1, \dot{p}_1, \ddot{p}_1]$ pe un interval de timp $T = t_1 - t_0$, este optimă din punctul de vedere al jerk-ului. Mai precis, polinoamele de ordinul cinci minimizează funcția de cost:

$$J_t(p(t)) = \int_{t_0}^{t_1} \ddot{p}^2(\tau) d\tau. \quad (\text{II.132})$$

În [221] este demonstrată următoarea propoziție: Având o stare inițială $P_0 = [p_0, \dot{p}_0, \ddot{p}_0]$ la momentul de timp t_0 și o stare finală $P_1 = [p_1, \dot{p}_1, \ddot{p}_1]$ la momentul de timp $t_1 = t_0 + T$, atunci soluția care minimizează funcția de cost

$$C = k_j J_t + k_t g(T) + k_p h(p_1) \quad (\text{II.133})$$

este de asemenea un polinom de ordinul cinci, unde funcțiile g și h sunt arbitrare, și $k_j, k_t, k_p > 0$.

Algoritmul poate fi separat în două părți, pentru generarea traiectoriilor la viteze mari și la viteze reduse. La viteze ridicate, deoarece se încearcă minimizarea jerk-ului, se alege ca stare inițială pentru optimizarea mișcării laterale $D_0 = [d_0, \dot{d}_0, \ddot{d}_0]$, care aparține traiectoriei calculate anterior, astfel încât să nu apară discontinuități între traiectorii. Pentru partea de optimizare, considerăm că starea finală $D_1 = [d_1, \dot{d}_1, \ddot{d}_1]$ cu $\dot{d}_1 = 0$ și $\ddot{d}_1 = 0$, deoarece vrem ca vehiculul să se deplaseze paralel cu traiectoria de referință.

Adițional, funcția de cost (II.133) este selectată

$$C_d = k_j J_t(d(t)) + k_t T + k_d d_1^2 \quad (\text{II.134})$$

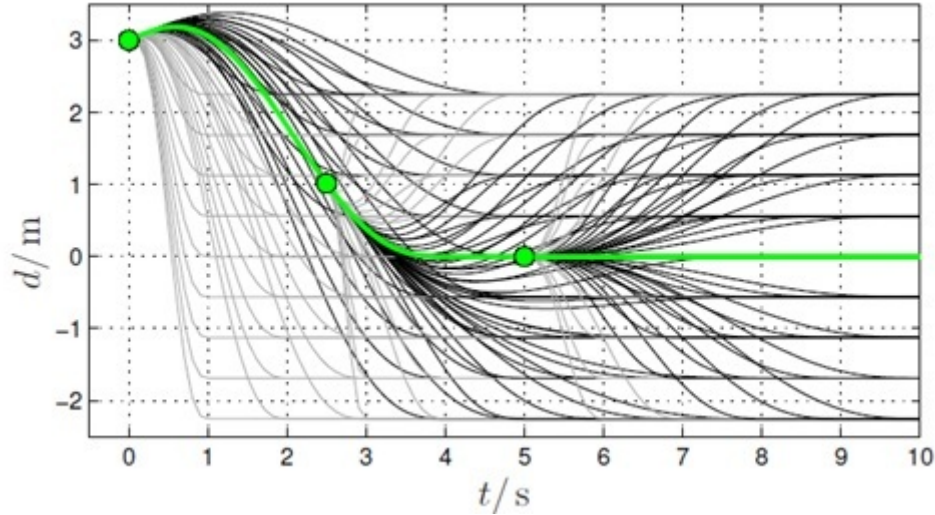


Figura II.44: Mișcarea laterală optimală rezultată din replanificarea ciclică.

astfel încât să fie penalizate soluțiile care sunt deplasate de referință în punctul final. Trebuie remarcat faptul că funcția de cost (II.134) (și cele ce urmează), nu depind de viteza vehiculului. După cum se știe din propoziția de mai sus, soluția care minimizează (II.134) trebuie să fie un polinom de ordinul cinci, iar acesta poate fi calculat ușor știind punctul de start și punctul de stop al traiectoriei. După ce polinomul a fost calculat, acesta va fi verificat împotriva coliziunilor și a restricțiilor dinamice. În cazul în care traiectoria se dovedește a fi invalidă din punct de vedere al restricțiilor, o altă traiectorie trebuie calculată și tot procesul repetat. De aceea, pentru a crește șansa găsirii unei traiectorii valide, un întreg set de puncte de stop va fi generat, calculând astfel pentru fiecare traiectorie un polinom care va fi verificat împotriva restricțiilor. Primul polinom valid va fi returnat și trimis către algoritmul de control ca traiectorie pe care vehiculul trebuie să o urmarească. După cum se poate observa din Fig. II.44, la momentul de timp $t = 0$, aproximativ toate traiectoriile posibile sunt luate în considerare. În pasul următor de planificare, o altă traiectorie optimală validă va fi selectată. A se observa faptul că, la fiecare planificare segmentul de traiectorie care rămâne de parcurs, este soluția optimală la următoarea planificare.

La viteze reduse, spre deosebire de cazul în care vehiculul circulă cu viteză ridicată, mișcarea laterală trebuie considerată depinzând de mișcarea longitudinală astfel:

$$\vec{x}(s(t), d(t)) = \vec{r}(s(t)) + d(s(t))\vec{n}_r(s(t)) \quad (\text{II.135})$$

Funcția de cost va fi obținută astfel:

$$C_d = k_j J_s(d(s)) + k_t S + k_d d_1^2 \quad (\text{II.136})$$

cu

$$S = s_1 - s_0, (\cdot)' = \frac{\partial}{\partial s}(\cdot) \quad (\text{II.137})$$

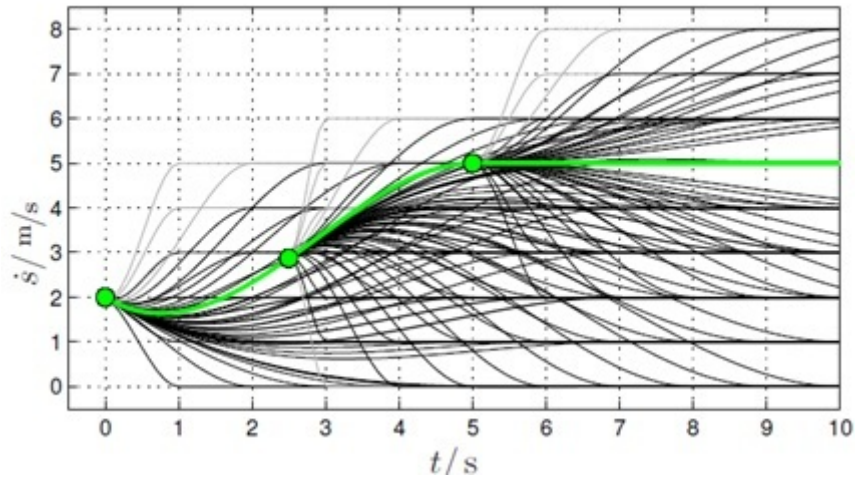


Figura II.45: Mișcarea longitudinală optimală rezultată din replanificarea ciclică.

și

$$J_s(d(s)) = \int_{s_0}^{s_1} d'''^2(\sigma) d\sigma \quad (\text{II.138})$$

Soluția obținută va fi descrisă tot de un polinom de ordinul cinci. Generarea setului de traiectorii se poate face similar ca și în cazul vitezelor ridicate.

În ceea ce privește deplasarea longitudinală, spre deosebire de alte metode unde durata de deplasare pe o anumită distanță era criteriul principal, metoda propusă se concentrează pe confort și siguranță în special la viteze ridicate. Din acest motiv, ca și criteriu de optimizare va fi luat în considerare jerk-ul longitudinal.

Traectoria longitudinală este astfel generată asemănător cu cea laterală, considerând starea de start a traiectoriei de forma $S_0 = [s_0, \dot{s}_0, \ddot{s}_0]$, până la anumite stări finale ale traiectoriei, date sub forma

$$S_1 = [s_1, \dot{s}_1, \ddot{s}_1] == [[s_{final}(T_j) + \Delta s_i], \dot{s}_{final}(T_j), \ddot{s}_{final}(T_j), T_j]$$

, așa cum este prezentat în Fig. II.45. În final, fiecare polinom generat va fi evaluat folosind funcția de cost

$$C_t = k_j J_t + k_t T + k_s [s_1 - s_d]^2 \quad (\text{II.139})$$

După ce traiectoriile laterale și longitudinale au fost separat generate, acestea trebuie să fie combinate astfel încât traiectoria dorită a vehiculului să țină cont de ambele dinamici (laterale și longitudinale). În cele ce urmează, traiectoriile laterale și cele longitudinale vor fi diferențiate prin notațiile *lat* și *long*.

Din cele două traiectorii generate vor fi calculate stările prin care vehiculul trebuie să treacă în reperul Cartezian, respectiv unghiul de orientare $\theta(t)$, curbura $\kappa(t)$, viteza $v(t)$, accelerația $a(t)$. Calculele necesare pentru a trece din reperul Frenet în reperul Cartezian, pot fi regăsite în [221].

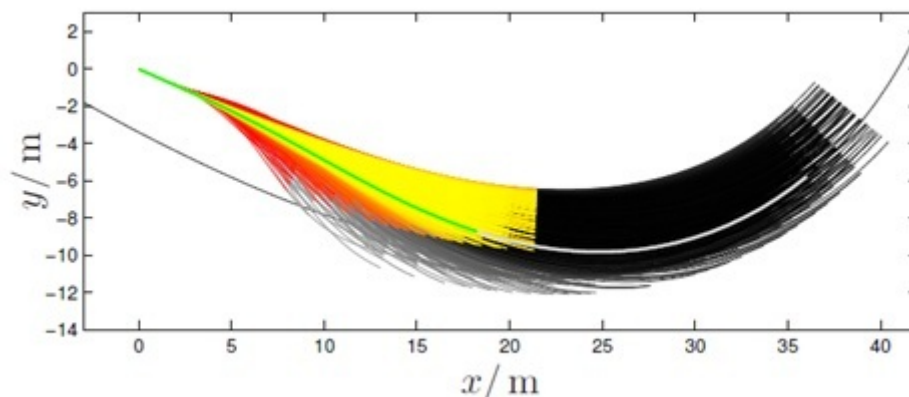


Figura II.46: Traiectorie rezultată în reperul Cartezian.

Pentru evitarea obstacolelor, o distanță de siguranță este luată în considerare, pe baza căreia se poate verifica fiecare traiectorie dacă se intersectează cu obiectele participante la trafic. În Fig. II.46, este reprezentată în reperul Cartezian o traiectorie finală rezultată.

În Figurile II.47, II.48 și II.49 este reprezentată o simulare a algoritmului de planificare și generare de traiectorii.

II.2.8 Controlul predictiv al lanțului de transmisie a puterii la autovehicule în rețea bazat pe restricții probabilistice

În ultimele decade rețelele de comunicații și aplicațiile acestora au trecut prin schimbări majore, ajungând astăzi să fie practic un standard în sistemele de control distribuite. Acest domeniu este în continuă dezvoltare, fiind de un real interes în aplicații pornind de la vehicule autonome militare, platforme de producție de dimensiuni mari și rafinării, până la robotică și medicină bazată pe telecomunicații. Sistemele care îmbină unități de calcul digitale și unități fizice sunt cunoscute sub denumirea de sisteme ciber-fizice (eng. Cyber-Physical Systems), și ele necesită o colaborare optimă între strategia de control (atât unitatea de calcul cât și algoritmul software) și procesul fizic reglat. Datorită evoluției sistemelor de control în rețea (eng. Networked Control Systems), s-a putut optimiza costul, instalarea și mentenanța, fiabilitatea și flexibilitatea sistemelor de reglare, dar în același timp, performanțele acestor sisteme pot fi alterate de neajunsul major al sistemelor de comunicații, și anume de întârzierile care pot apărea în transmisia datelor sau chiar de pierderile de date [33]. Aceste întârzieri sunt variabile în timp și nu pot fi anticipate, având caracteristici stohastice putând destabiliza bucla de control, ceea ce face necesară analiza și considerarea întârzierilor încă din faza de proiectare a sistemului de reglare.

Practic, din cauza întârzierilor care apar pe linia de comunicație, care pot fi considerate perturbații ce nu pot fi prevăzute, performanțele cerute pentru sistemul de control nu mai pot fi îndeplinite cu probabilitate unu. De aceea, performanțele trebuie definite în termeni de probabilitate. Controlul predictiv permite definirea cerințelor în formă probabilistică prin introducerea restricțiilor probabilistice, astfel algoritmul fiind extins la control predictiv bazat pe restricții probabilistice (eng. Chance-Constrained Model Predictive Control).

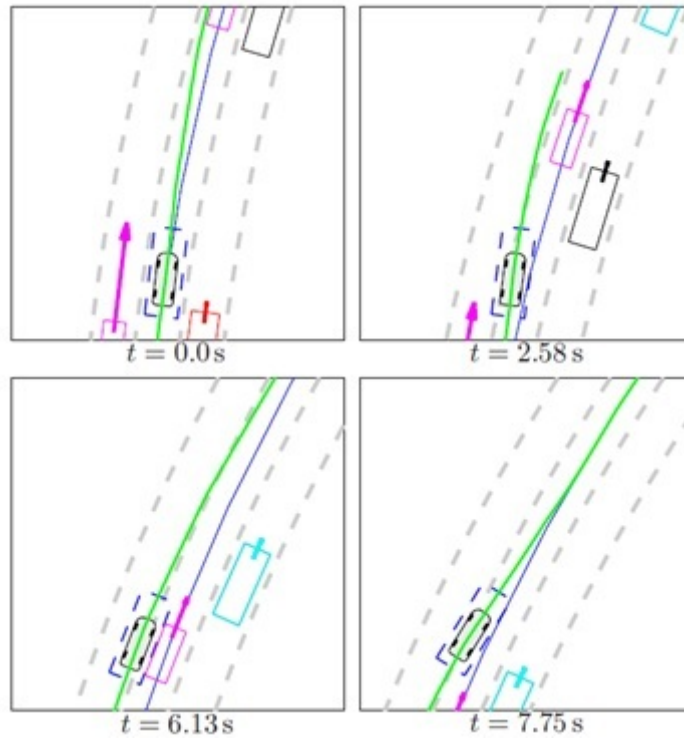


Figura II.47: Simularea unui scenariu de conducere automată pe autostradă I.

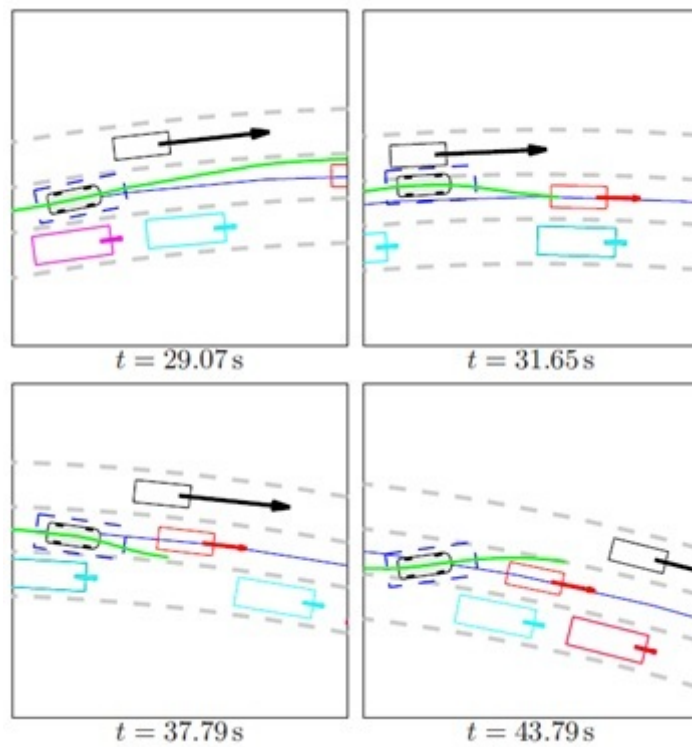


Figura II.48: Simularea unui scenariu de conducere automată pe autostradă II.

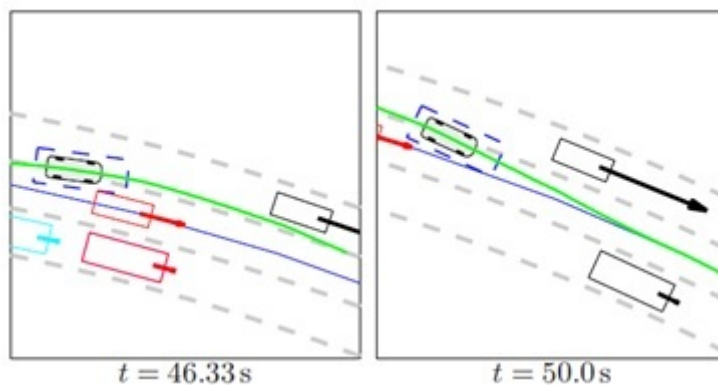


Figura II.49: Simularea unui scenariu de conducere automată pe autostradă III.

Această variantă de control predictiv a fost studiată intens, și diferite rezultate au fost publicate [224–235], atât folosind modele liniare ale procesului reglat [224–229, 231–235], cât și modele liniarizate pe porțiuni, unde modelele se schimbă în funcție de perturbații [230].

Perturbațiile considerate sunt aditive [224–229, 231–234] sau multiplicative [230], iar sistemele au caracteristici diferite - distribuția de probabilitate a ieșirii este cunoscută [224, 225, 227], distribuție gaussiană a perturbației [226, 228, 229, 233], covarianță și medie cunoscută pentru intrări și ieșiri [232, 234, 235], distribuție de probabilitate cunoscută [229] sau observabilă la fiecare pas [233] a perturbației. Pornind de la aceste caracteristici, restricțiile probabilistice impuse intrărilor [228, 229, 232, 234, 235], stărilor [226, 228–230, 232, 234, 235] și ieșirilor [224, 225, 227] sunt transformate în restricții deterministe impuse intrării procesului.

În ceea ce privește tipul restricțiilor probabilistice, în literatura de specialitate se pot găsi două metode: restricții probabilistice individuale [224–227, 230, 232–235] și restricții probabilistice unificate [228, 231]. În implementarea controlului predictiv, a fost considerată varianta cu reacție după ieșire [224, 225, 227, 232, 234], și reacție după stare [226, 229–231, 233, 235].

Printre aplicațiile în care a fost aplicat controlul predictiv cu restricții probabilistice, se pot găsi procese de distilare [224], controlul rezervoarelor [225], controlul traficului în rețea [226], controlul morilor [227] și controlul rețelelor de apă potabilă [231].

Notații și definiții. \mathbb{R} , \mathbb{R}_+ , \mathbb{Z} and \mathbb{Z}_+ sunt numere reale, reale pozitive, întregi și întregi pozitive. $\mathbb{Z}_{\geq c_1}$ și $\mathbb{Z}_{(c_1, c_2]}$ reprezintă seturile $\{k \in \mathbb{Z}_+ \mid k \geq c_1\}$ și $\{k \in \mathbb{Z}_+ \mid c_1 < k \leq c_2\}$, respectiv, pentru $c_1, c_2 \in \mathbb{Z}_+$. $\|\cdot\|_\infty$ reprezintă norma infinit; având $x \in \mathbb{R}^n$ and $c \in \mathbb{R}$, prin definiția normei infinit, pentru ca $\|x\|_\infty \leq c$ să fie îndeplinit, este necesar și suficient ca $\pm[x]_j \leq c$ pentru toate $j \in \mathbb{Z}_{[1, n]}$. I_n este matricea identitate de ordin n și M' reprezintă transpusa matricii M . $Pr[A]$ este probabilitatea unui eveniment A și $p(x)$ reprezintă densitatea de probabilitate a lui x . \bar{x} reprezintă valoarea medie a variabilei aleatoare x și S_x reprezintă covarianța.

II.2.8.1 Formularea problemei

Fie un sistem liniar invariant în timp:

$$x_{k+1} = Ax_k + Bu_k + B_w w_k, \quad (\text{II.140})$$

unde $x_k \in \mathbb{R}^n$ este starea sistemului, $u_k \in \mathbb{R}^m$ este intrarea, $w_k \in \mathbb{R}^p$ este zgomotul, cu k momentul de timp discret, iar $A \in \mathbb{R}^{n \times n}$ și $B \in \mathbb{R}^{n \times m}$ sunt matricile sistemului, și $B_w \in \mathbb{R}^{n \times p}$ este matricea perturbației. Vectorul de zgomot poate modela perturbații, erori de modelare, zgomotul de măsură, și perturbații introduse de rețeaua de comunicații. Se presupune ca perturbația w este generată de un proces gaussian independent și uniform distribuit și starea inițială x_0 este o variabilă gaussiană aleatoare; aceste variabile aleatoare nu sunt corelate.

În controlul optimal cu orizont finit, problema de optimizare se rezolvă de-a lungul unui orizont finit $k = 0$ până la $k = N - 1$, unde N se numește și orizont de predicție. Pentru conveniență, variabilele de-a lungul orizontului de predicție se pot scrie [228]:

$$\mathbb{X} = \begin{bmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{bmatrix}, \mathbb{U} = \begin{bmatrix} u_0 \\ u_1 \\ \vdots \\ u_{N-1} \end{bmatrix}, \mathbb{W} = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_N \end{bmatrix}. \quad (\text{II.141})$$

În forma de mai sus, modelul dinamic devine

$$\mathbb{X} = G_{xx}x_0 + G_{xu}\mathbb{U} + G_{xw}\mathbb{W}, \quad (\text{II.142})$$

unde matricile G_{xx} , G_{xu} și G_{xw} sunt calculate prin înmulțiri repetate ale sistemului (II.140). Având o funcție de cost $J(\mathbb{X}, \mathbb{U})$ convexă în \mathbb{X} și \mathbb{U} , și un set convex de restricții asupra comenzii F_U , și un politop convex de restricții asupra stărilor F_X , problema de control optimal cu restricții probabilistice se poate formula astfel

Problema II.2.4 *Problema de control cu restricții probabilistice.*

Minimizează $J(\mathbb{X}, \mathbb{U})$

Cu: $\mathbb{U} \in F_U$

$$Pr[\mathbb{X} \notin F_X] \leq \delta$$

$$\mathbb{X} = G_{xx}x_0 + G_{xu}\mathbb{U} + G_{xw}\mathbb{W}.$$

Cu alte cuvinte, semnalul de control trebuie să minimizeze funcția de cost, astfel încât stările sistemului ies din regiunea fezabilă cu o probabilitate mai mică decât δ .

Problema II.2.4 este complicată din cauza restricției probabilistice $Pr[\mathbb{X} \notin F_X] \leq \delta$. Din această restricție se pot extrage trei aspecte [228]:

- distribuția lui \mathbb{X} ca funcție de semnalul de control \mathbb{U} trebuie determinată;
- o integrală multidimensională a acestei distribuții trebuie calculată;
- optimizarea trebuie făcută ținând cont de restricții de-a lungul acestei integrale.

Primul aspect este rezolvat datorită liniarității modelului și a zgomotului gaussian. În acest caz, stările sistemului \mathbb{X} sunt variabile aleatoare gaussiene cu valoare medie și distribuție date de [228]

$$\begin{aligned}\bar{\mathbb{X}} &= G_{xx}\bar{x}_0 + G_{xu}\mathbb{U} + G_{xw}\bar{\mathbb{W}} \\ S_{\mathbb{X}} &= G_{xx}S_{x_0}G'_{xx} + G_{xw}S_{\mathbb{W}}G'_{xw}.\end{aligned}\quad (\text{II.143})$$

Regiunea fezabilă convexă F_X poate fi definită ca o unificare a M restricții inegalitate liniare

$$F_X \triangleq \bigcap_{i=1}^M \{\mathbb{X} : a'_i \mathbb{X} \leq b_i\}, \quad (\text{II.144})$$

unde a_i este o matrice și b_i este un vector de dimensiuni corespunzătoare. Deci, o aproximare a problemei de control cu restricții probabilistice II.2.4 poate fi reformulată ca:

Problema II.2.5 *Problema de control conservativă cu restricții probabilistice.*

$$\begin{aligned}\text{Minimize} \quad & J(\mathbb{X}, \mathbb{U}) \\ \text{Subject to:} \quad & \mathbb{U} \in F_U \\ & Pr[a'_i \mathbb{X} > b_i] \leq \epsilon_i, \forall i \\ & \sum_{i=1}^M \epsilon_i \leq \delta \\ & \bar{\mathbb{X}} = G_{xx}\bar{x}_0 + G_{xu}\mathbb{U} + G_{xw}\bar{\mathbb{W}}.\end{aligned}$$

Lema II.2.6 *O soluție fezabilă a Problemei II.2.5 (problema conservativă) este o soluție fezabilă a Problemei II.2.4 [228].*

În această secțiune, o aproximare a Problemei II.2.5 este folosită în locul Problemei II.2.4. Rezolvând Problema II.2.5, optimizarea se face explicit calculând probabilitatea ca fiecare restricție individuală să fie încălcată, notată ϵ_i .

Diferența principală dintre Problema II.2.5 și Problema II.2.4 este că Problema II.2.5 nu mai necesită rezolvarea integralelor multivariate. În schimb, are M restricții *univariate* integrale. Pentru a observa aceasta, se definește $y_i \triangleq a'_i x$ și se ține cont că y_i este o variabilă aleatoare gaussiană, cu valoare medie și varianță date de

$$\begin{aligned}\bar{y}_i &= a'_i G_{xx}\bar{x}_0 + a'_i G_{xu}\mathbb{U} + a'_i G_{xw}\bar{\mathbb{W}} \\ S_{y_i} &= a'_i G_{xx}S_{x_0}G'_{xx}a_i + a'_i G_{xw}S_{\mathbb{W}}G'_{xw}a_i.\end{aligned}\quad (\text{II.145})$$

Probabilitatea ca fiecare restricție să fie încălcată este definită ca

$$\begin{aligned}Pr[a'_i \mathbb{X} > b_i] &= Pr[y_i > b_i] = \\ &= \frac{1}{\sqrt{2\pi S_{y_i}}} \int_{b_i}^{\infty} e^{-\frac{(y_i - \bar{y}_i)^2}{2S_{y_i}}} dy_i.\end{aligned}\quad (\text{II.146})$$

Deoarece (II.146) este o integrală univariată, poate fi exprimată ca *cdf*

$$\begin{aligned}Pr[a'_i \mathbb{X} > b_i] &= \frac{1}{\sqrt{2\pi}} \int_{\frac{b_i - \bar{y}_i}{\sqrt{S_{y_i}}}}^{\infty} e^{-\frac{z^2}{2}} dz \\ &= 1 - cdf\left(\frac{b_i - \bar{y}_i}{\sqrt{S_{y_i}}}\right),\end{aligned}\quad (\text{II.147})$$

unde cdf este distribuția gaussiană standard cumulativă.

$$cdf(x) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^x e^{-\frac{z^2}{2}} dz. \quad (\text{II.148})$$

Folosind (II.147), Problema II.2.5 se poate rescrie ca [228].

Problema II.2.7 Problema conservativă de control cu restricții probabilistice - forma integrală.

$$\begin{aligned} \text{Minimizeaza} \quad & J(\bar{\mathbb{X}}, \mathbb{U}) \\ \text{Tinand cont de:} \quad & \mathbb{U} \in F_U \\ & 1 - cdf\left(\frac{b_i - \bar{y}_i}{\sqrt{S_{y_i}}}\right) \leq \epsilon_i, \forall i \\ & \bar{y}_i = a_i' G_{xx} \bar{x}_0 + a_i' G_{xu} \mathbb{U} + a_i' G_{xw} \bar{\mathbb{W}} \\ & S_{y_i} = a_i' G_{xx} S_{x_0} G_{xx}' a_i + a_i' G_{xw} S_{\mathbb{W}} G_{xw}' a_i \\ & \sum_i^M \epsilon_i \leq \delta. \end{aligned}$$

Este prezentată o aproximare conservativă a problemei controlului cu restricții probabilistice. O soluție fezabilă a acestei probleme este o soluție fezabilă a problemei originale. Problema conservativă este convexă, rezultând că pot fi folosiți algoritmi numerici de minimizare neliniari.

$$\text{Se observă că } 1 - cdf\left(\frac{b_i - \bar{y}_i}{\sqrt{S_{y_i}}}\right) \leq \epsilon_i, \forall i \text{ poate fi rescris } \frac{b_i - \bar{y}_i}{\sqrt{S_{y_i}}} \geq icdf(1 - \epsilon_i), \forall i.$$

Remarca II.2.8 Funcția de distribuție invers cumulativă poate fi măriginită folosind aproximări liniare așa cum se va arăta în continuare.

Folosind Remarca II.2.8, Problema II.2.7 poate fi rescrisă după cum urmează.

Problema II.2.9 Problema conservativă de control cu restricții probabilistice - forma liniară.

$$\begin{aligned} \text{Minimizeaza} \quad & J(\bar{\mathbb{X}}, \mathbb{U}) \\ \text{Tinand cont de:} \quad & \mathbb{U} \in F_U \\ & \frac{b_i - \bar{y}_i}{\sqrt{S_{y_i}}} \geq h(1 - \epsilon_i), \forall i \\ & \bar{y}_i = a_i' G_{xx} \bar{x}_0 + a_i' G_{xu} \mathbb{U} + a_i' G_{xw} \bar{\mathbb{W}} \\ & S_{y_i} = a_i' G_{xx} S_{x_0} G_{xx}' a_i + a_i' G_{xw} S_{\mathbb{W}} G_{xw}' a_i \\ & \sum_i^M \epsilon_i \leq \delta, \end{aligned}$$

unde $h(\cdot)$ este o funcție liniară.

În Fig. II.50 se prezintă aproximări liniare posibile ale distribuției cumulative inverse necesare Problemei II.2.9. În faza de proiectare a algoritmului de control, trebuie aleasă o anumită aproximare liniară, în funcție de aplicație.

Lema II.2.10 O soluție fezabilă a Problemei II.2.9 (problema liniară conservativă) este o soluție fezabilă a Problemei II.2.7.

Demonstrația lemei II.2.10 este dată în [236] și este omisă în acest text.

În această secțiune, se rezolvă Problema II.2.9, fiind liniară în variabilele de optimizare.

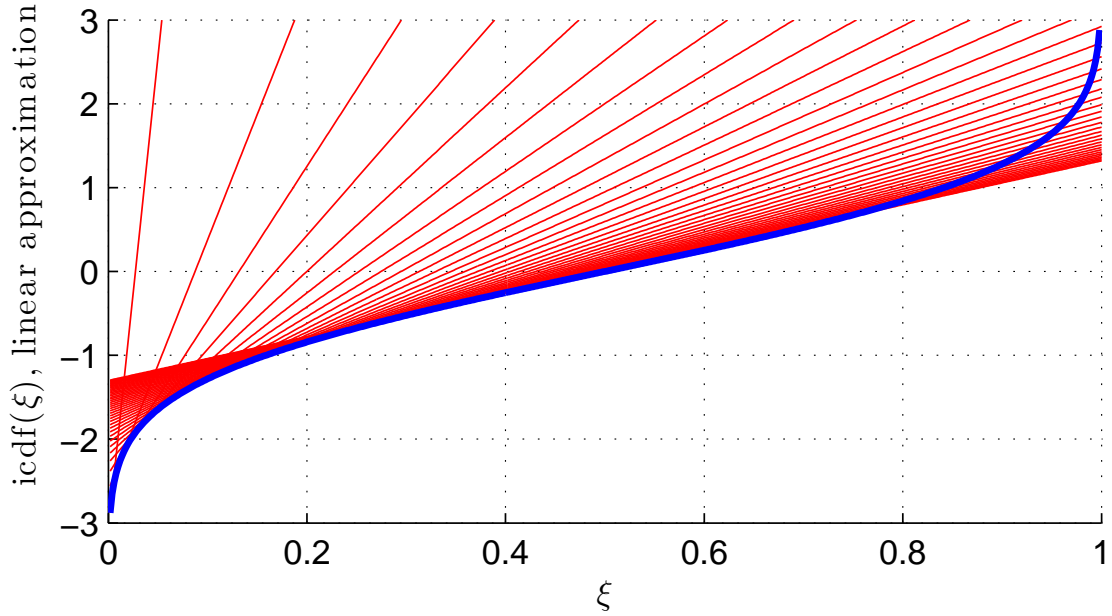


Figura II.50: Funcția de distribuție cumulativă inversă (linie îngroșată) și aproximațiile liniare (tangente la $icdf$).

Remarca II.2.11 Lema II.2.10 asigură că o soluție optimală a Problemei II.2.9 este o soluție sub-optimală a Problemei II.2.4.

II.2.8.2 Modelarea întârzierilor introduse de rețeaua de comunicații

În această subsecțiune, se prezintă modelarea întârzierilor care pot apărea în rețeaua de comunicație.

II.2.8.3 Configurația sistemului de control în rețea

Fie sistemul de control în rețea ilustrat în Fig. II.51 care este alcătuit din două părți: sistemul fizic, un senzor (S), regulatorul, actuatorul (A) și o rețea de comunicații între regulator și proces. x este starea sistemului, u este intrarea sistemului și x^r este referința impusă.

Remarca II.2.12 Așa cum este arătat în [66] întârzierile care apar pe linia de comunicații pe calea de reacție, pot fi văzute ca întârzieri care apar pe calea directă ca în Fig. II.51 dacă anumite condiții sunt îndeplinite.

Întârzierile care apar pe linia de comunicație sunt considerate ca fiind multiple de perioada de eșantionare $T_s \in \mathbb{R}_+$, deci, $d_k = \lceil \tau_k / T_s \rceil \in \mathbb{Z}_{\geq 0}$, unde τ_k este întârzierea pe calea directă (de la regulator la elementul de execuție) la momentul discret de timp $k \in \mathbb{Z}_+$. Fie \bar{d} întârzierea maximă pe calea directă, exprimată ca un număr de perioade de eșantionare $\bar{d} = \lceil \tau^{max} / T_s \rceil$, deci $d_k \in \mathbb{Z}_{[0, \bar{d}]}$.

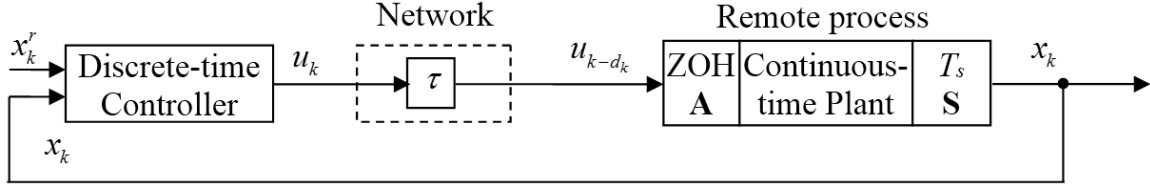


Figura II.51: Sistem de control în rețea cu întârzieri.

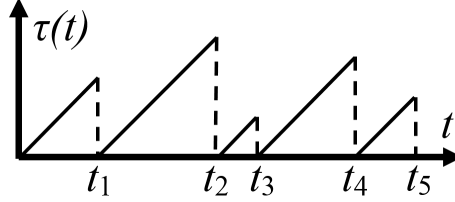


Figura II.52: Exemplu de întârzieri descriind o funcție extrapolator de ordin 0 cu o realizare aperiodică.

Canal de comunicații stohastic Întârzierea pe calea directă $\tau(t)$ are o formă tip dinte de fierastrău, așa cum se arată în Fig. II.52. Întârzierile satisfac $\dot{\tau}(t) = 1$ aproape întotdeauna cu discontinuități, devenind nule la anumite momente de timp (cum este arătat în Fig. II.52).

$$0 < t_1 < t_2 < \dots, \quad (\text{II.149})$$

care corespund cu momentele de timp de sosire a unui proces Poisson omogen cu rata λ . Fie $t_0 = 0$, momentele de timp între sosiri $\delta_k := t_{k+1} - t_k, k \geq 0$, sunt independente și distribuite identic, și densitatea de probabilitate între momentele de sosire δ_k este dată de [237]

$$\Pr[\delta_k = t] = \begin{cases} \lambda e^{-\lambda t}, & \text{if } 0 \leq t < \tau^{max}, \\ \lambda \int_{\tau^{max}}^{\infty} e^{-\lambda t} dt, & \text{if } t = \tau^{max}. \end{cases} \quad (\text{II.150})$$

Modelul sistemului de control în rețea Modelul continuu al procesului cu întârzieri pe intrare este dat de [34]

$$\begin{aligned} \dot{x}(t) &= A_c x(t) + B_c u^*(t), \\ u^*(t) &= u_k, \forall t \in [t_k + d_k, t_{k+1} + d_{k+1}], \end{aligned} \quad (\text{II.151})$$

unde $t_k = kT_s, k \in \mathbb{Z}_+$ și $u^*(t) = u_0$ pentru $t \in [0, d_0]$ cu $u_0 \in \mathbb{R}$ un vector constant dat. $u_k \in \mathbb{R}$ este semnalul de control calculat la momentul de timp $t = t_k$.

Presupunând că $u_k = \psi_k$ pentru $k \in \mathbb{Z}_{[-\bar{d}-1, -1]}$ cu $\psi_{[-\bar{d}-1, -1]}$ un vector predeterminat, modelul discret devine

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k + B \left(\Delta_0(d_k)(u_{k-1} - u_k) \right. \\ &\quad + \Delta_1(d_k)(u_{k-2} - u_{k-1}) + \dots \\ &\quad \left. + \Delta_{\bar{d}}(d_k)(u_{k-\bar{d}-1} - u_{k-\bar{d}}) \right), \end{aligned} \quad (\text{II.152})$$

unde $A = e^{AcT_s}$, $B = \int_0^{T_s} e^{Ac(T_s-\theta)} d\theta B_c$,

$$\Delta_i(d_k) := \begin{cases} 0, & d_{k-i} - i \leq 0 \\ 1, & 1 \leq d_{k-i} - i \end{cases} \quad (\text{II.153})$$

cu $k \in \mathbb{Z}_+$ and $i \in \mathbb{Z}_{[0, \bar{d}]}$.

Modelul discret al sistemului cu întârzieri pe intrare (II.172) poate fi rescris ca

$$x_{k+1} = Ax_k + Bu_k + Bu_k^d, \quad (\text{II.154})$$

unde termenul

$$u_k^d = \sum_{i=0}^{\bar{d}} \Delta_i(d_k)(u_{k-i-1} - u_{k-i}) \quad (\text{II.155})$$

este definit ca *perturbație introdusă de rețea*. De observat că (II.174) devine (II.140) când $B_w w_k \equiv Bu_k^d$ este setat.

Presupunerea II.2.13 *Perturbația introdusă de rețea* definită de (II.155) formează un proces aleator independent și distribuit identic.

Remarca II.2.14 Simularea prezentată în secțiunea II.2.8.5 arată că presupunerea gaussiană II.2.13 este într-adevăr satisfăcută.

Controlul predictiv al sistemului în rețea Pentru implementarea controlului predictiv propus, sistemul în rețea (II.174) este conectat cu Problema II.2.9.

Presupunerea II.2.15 În Problema II.2.9, funcția de cost este dată de

$$J(\bar{X}, \mathbb{U}) = \|Q\bar{X}\|_\infty + \|R\mathbb{U}\|_\infty, \quad (\text{II.156})$$

cu Q și R fiind matrici de dimensiuni corespunzătoare.

Algoritmul predictiv funcționează după principiul orizontului alunecător, adică funcția de cost $J(\cdot, \cdot)$ din Problema II.2.9 trebuie minimizată la fiecare moment de esantionare $k \in \mathbb{Z}_{[0, N-1]}$. Intrarea curentă u_k în (II.174) este obținută determinând secvența de intrare $\{\hat{u}_k, \dots, \hat{u}_{k+N-1}\}$ care rezolvă Problema II.2.9, și alegând $u_k = \hat{u}_k$. Restul elementelor din secvența de control nu mai sunt utilizate, iar acest proces iterativ este repetat la fiecare moment de esantionare.

II.2.8.4 Modelarea lanțului de transmisie al vehiculului

În această secțiune se prezintă modelarea lanțului de transmisie a unui vehicul, care este folosit în Secțiunea II.2.8.5 pentru proiectarea sistemului de reglare [34].

Modelul cu două mase ilustrat în Fig. II.53, care reprezintă un lanț de transmisie simplificat cu ambreajul cuplat, are două inerții, una pentru cutia de viteze J_{eg} și una reprezentând efectele vehiculului și a roților J_v conectate printr-un ax flexibil. Motorul generează cuplul T_e , care este considerat semnalul de control și este considerat disponibil la cererea unității

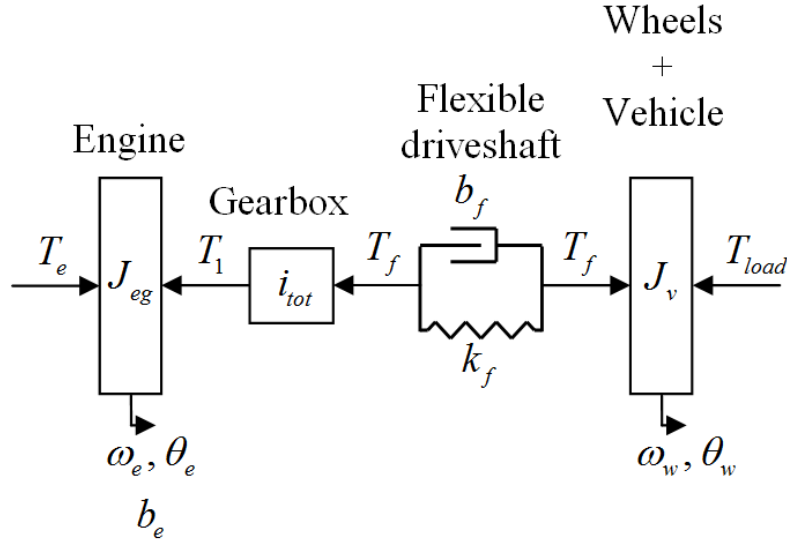


Figura II.53: Reprezentarea schematica simplificata a lantului de transmisie.

de control a motorului și este obținut prin modularea timpului scânteii și a fluxului de aer și este transmis către roți.

Flexibilitatea din model este dată de flexibilitatea osiei și a axului propulsor, care transmite cuplul primit de la cutia de viteze către roți. Roțile motoare sunt ultimele componente ale lanțului de transmisie a puterii, care are rolul de a pune vehiculul în mișcare, învingând forțele de frecare, care apar datorită contactului cauciucului cu suprafața de rulare, și a forței rezistente aerodinamice.

Dinamica vitezei unghiulare a motorului este dată de

$$J_{eg}\dot{\omega}_e = T_e - b_e\omega_e - T_f/i_{tot}, \quad (\text{II.157})$$

unde $J_{eg} := J_e + \frac{J_g}{i_{tot}^2}$ este inerția echivalentă motor-cutie de viteze, unde J_e este inerția motorului, J_g este inerția cutiei de viteze exprimate din punctul de vedere al axului propulsor, cu $i_{tot} := i_g i_f$ raportul de transmisie, ω_e este viteza unghiulară a motorului, b_e este coeficientul de frecare vâscoasă a motorului și T_f este cuplul axului propulsor flexibil.

Cuplul axului propulsor flexibil este dat ca

$$T_f = b_f (\omega_e/i_{tot} - \omega_w) + k_f (\theta_e/i_{tot} - \theta_w), \quad (\text{II.158})$$

unde b_f și k_f sunt coeficientul de frecare vâscoasă și factorul de elasticitate ale axului propulsor, θ_e este unghiul motorului, θ_w și ω_w sunt unghiul și viteza unghiulară ale roților.

În mod analog, corpul vehiculului poate fi modelat astfel

$$J_v\dot{\omega}_w = T_f - T_{load}, \quad (\text{II.159})$$

unde cauciucul a fost considerat ca un element care se rotește fara alunecare, deci $v_v = r_w\omega_w$, unde v_v este viteza vehiculului și r_w este raza efectivă a roții. Inerția vehiculului poate fi obținută prin adunarea inerției roții la inerția echivalentă a vehiculului obținută din masa acesteia, adică

$$J_v = J_w + m_v r_w^2, \quad (\text{II.160})$$

unde m_v este masa vehiculului și J_w este inerția roții.

Cuplul de sarcina este modelat astfel

$$T_{load} = T_{airdrag} + T_{roll} + T_{grade}, \quad (\text{II.161})$$

unde $T_{airdrag}$ este cuplul rezistent aerodinamic, ce acționează asupra corpului vehiculului, T_{roll} este cuplul generat de cauciucuri și T_{grade} este cuplul generat de panta drumului. Pentru detalii cititorii interesați pot face referire la [34].

Modelul continuu Considerând stările sistemului viteza unghiulară a motorului, viteza unghiulară a roții și torsiunea axului propulsor

$$x_1^m = \omega_e, \quad x_2^m = \omega_w, \quad x_3^m = \frac{\theta_e}{i_{tot}} - \theta_w, \quad (\text{II.162})$$

modelul intrare-stare-ieșire devine

$$\dot{x}^m(t) = A_c^m x^m(t) + B_c^m u^m(t) + f_c^m, \quad (\text{II.163})$$

unde $x^m = \begin{pmatrix} x_1^m & x_2^m & x_3^m \end{pmatrix}^\top$ este starea sistemului, $u^m \in \mathbb{R}$ este intrarea, $f_c^m = \begin{pmatrix} 0 \\ -\frac{T_{load}}{J_v} \\ 0 \end{pmatrix}$, $B_c^m = \begin{pmatrix} \frac{1}{J_{eg}} \\ 0 \\ 0 \end{pmatrix}$ și $A_c^m = \begin{pmatrix} -\frac{b_e}{J_{eg}} - \frac{b_f}{i_{tot}^2 J_{eg}} & \frac{b_f}{i_{tot} J_{eg}} & -\frac{k_f}{i_{tot} J_{eg}} \\ \frac{b_f}{i_{tot} J_v} & -\frac{b_a + b_f}{J_v} & \frac{k_f}{J_v} \\ \frac{1}{i_{tot}} & -1 & 0 \end{pmatrix}$.

Cuplul motorului este restricționat astfel

$$0 \leq u^m(t) \leq T_e^{max}, \quad (\text{II.164})$$

unde T_e^{max} este cuplul maxim care poate fi generat de motor.

Viteza unghiulară a motorului este restricționată astfel

$$\omega_e^{min} \leq x_1^m(t) \leq \omega_e^{max}, \quad (\text{II.165})$$

unde ω_e^{min} și ω_e^{max} sunt viteza minimă (repaus) și respectiv viteza maximă a motorului.

Se știe că oscilațiile lanțului de transmisie produc uzură mecanică și disconfort [34]. Viteza de torsiune, care este corelată în mod direct cu oscilațiile lanțului de transmisie, poate fi mărginită superior astfel

$$Pr[x_3^m(t) \geq x_3^{max}] \leq \delta, \quad (\text{II.166})$$

unde x_3^{max} reprezintă limita superioară dorită.

Obiectivul controlului este de a obține o valoare dorită a vitezei roții, $\omega_w^r = x_2^r$, ajungând la această valoare cu un regim tranzitoriu cât mai scurt posibil și suprareglare minimă. A se observa că pentru o valoare dorită a vitezei unghiulare a roții $x_2^{ss} \triangleq x_2^r$, se poate obține o valoare de regim staționar a vitezei unghiulare a motorului x_1^{ss} , vitezei de torsiune x_3^{ss} și a cuplului motorului u^{ss} prin alegerea derivatelor egale cu zero în (II.163).

Pentru a facilita implementarea sistemului de control predictiv cu restricții probabilistice se face o schimbare de coordonate a (II.163) astfel

$$x(t) = x^m(t) - x^{ss}(t), \quad u(t) = u^m(t) - u^{ss}(t), \quad (\text{II.167})$$

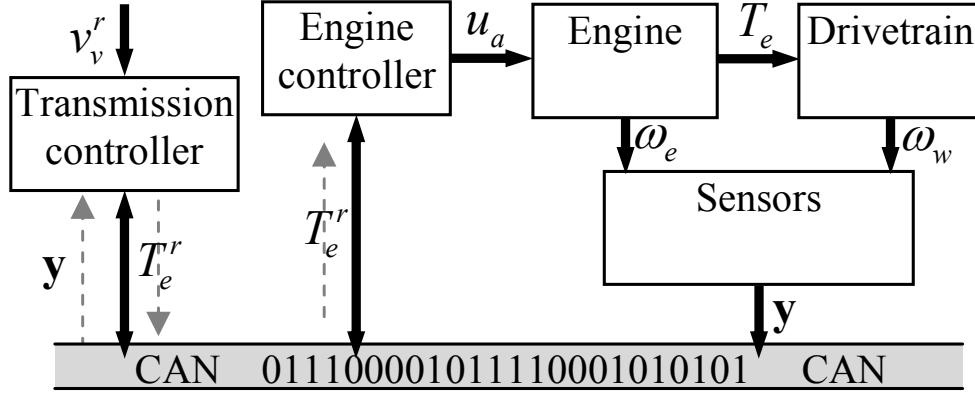


Figura II.54: Arhitectura sistemului de control

unde $x^{ss} = (x_1^{ss} \ x_2^{ss} \ x_3^{ss})^\top$, ceea ce duce la următoarea formă a sistemului

$$\dot{x}(t) = A_c x(t) + B_c u(t), \quad (\text{II.168})$$

unde A_c și B_c sunt matricile transformate ale sistemului.

Modelul discret Discretizând modelul (II.168) se obține

$$x_{k+1} = A x_k + B u_k, \quad (\text{II.169})$$

pentru $k \in \mathbb{Z}_+$, unde A și B sunt matricile sistemului discret x_k și u_k sunt starea și intrarea sistemului la momentul discret de timp $k \in \mathbb{Z}_+$.

Folosind (II.167) și (II.169), restricțiile date în (II.164), (II.165) și (II.166) pot fi convertite astfel:

$$\begin{aligned} x_{1,k} &\in [\underline{b}^{x_1}, \bar{b}^{x_1}], x_{2,k} \in [\underline{b}^{x_2}, \bar{b}^{x_2}], \\ Pr[x_{3,k} \geq \bar{b}^{x_3}] &\leq \delta, \\ u_k &\in [\underline{b}^u, \bar{b}^u], \Delta u_k \in [\underline{b}^{\Delta u}, \bar{b}^{\Delta u}], \end{aligned} \quad (\text{II.170})$$

pentru restricții definite adecvat, unde $\Delta u_k = u_k - u_{k-1}$.

Obiectivul de control poate fi formulat ca stabilizarea sistemului în punctul de echilibru zero al sistemului (II.169), respectând restricțiile (II.170).

Arhitectura sistemului de control al lanțului de transmisie Structura de control în rețea propusă în această secțiune, prezentată în Fig. II.54, funcționează după cum urmează:

- senzorii măsoară ieșirile sistemului și trimit semnalele către regulator prin CAN;
- regulatorul primește semnalele de la senzor și referința $v_v^r = r_w \omega_w^r$ și calculează cuplul dorit, ținând cont de restricții și de întârzieri; ω_w^r este viteza unghiulară dorită a roților;
- semnalul de control T_e^r este trimis către motor prin CAN;

- unitatea de control a motorului acționează scânteile și fluxul de aer (prin u_a) după cum este cerut de regulatorul lanțului de transmisie.

În Fig. II.54 liniile punctate reprezintă direcția mesajelor trimise la și de la reglatoare.

În această structură elementul de execuție este considerat ca fiind motorul și se presupune că cuplul dorit este obținut instantaneu. Aceasta poate fi realizată prin crearea unei rezerve de cuplu prin schimbarea timpului scânteii astfel încât cuplul să poată fi modificat rapid (Cairano 2010 Book), și poate fi menținută prin adaptarea corespunzătoare a fluxului de aer.

Comparând Fig. II.54 cu Fig. II.51 se poate observa că

- regulatorul transmisiei reprezintă regulatorul discret din Fig. II.51, care este implementat folosind controlul predictiv cu restricții probabilistice propus;
- rețeaua de comunicație CAN reprezintă rețeaua de comunicație din Fig. II.51 pentru care Remarca II.2.12 este îndeplinită;
- unitatea de control a motorului, motorul, și axul propulsor reprezintă sistemul fizic din Fig. II.51.

În consecință, considerând structura de control în rețea a lanțului de transmisie descris, știind că întârzierile care pot apărea pe linia CAN au comportamentul stohastic definit prin modelul descris în subsecțiunea II.2.8.3, modelul în rețea al vehiculului descris de (II.169) poate fi rescris ca în (II.174), unde este scos clar în evidență *networked-induced disturbance*.

II.2.8.5 Rezultate de simulare

În această secțiune sunt prezentate rezultate de simulare a structurii de control propuse pentru lanțul de transmisie a vehiculului din Fig. II.54. Modelul discret (II.174) a fost implementat în Matlab, cu o perioadă de eșantionare $T_s = 0.001s$ și valorile parametrilor utilizați în simulare sunt date în [34].

Obiectivul controlului este de a aduce viteza mașinii la o valoare dorită, având un regim tranzitoriu cât mai scurt, dar în același timp să crească nivelul de confort al pasagerilor prin reducerea oscilațiilor care apar când vehiculului este supus accelerațiilor.

Presupunerea II.2.13 cere ca perturbația indusă în rețeaua de comunicații u_k^d să fie o variabilă aleatoare independentă și identic distribuită. Pentru a verifica validitatea acestei presupunerii, întârzierile pe linia de comunicații au fost simulate folosind distribuția descrisă în Secțiunea II.2.8.3—un milion de puncte au generat întârzierile u_k^d cu $\lambda = 1$. Acestea se pot vedea în Fig. II.55. Deci, (II.169) a fost considerată în configurația (II.140).

Revenind la Problema II.2.9, aceasta cere o aproximare liniară a funcției de distribuție cumulativă inversă. Pentru ca acest lucru să fie îndeplinit, a fost considerată aproximarea liniară $h(1 - \varepsilon_i) = 2.38 - 4\varepsilon_i$, care este validă pentru $\varepsilon_i \in [0.01, 1)$, așa cum se poate vedea în Fig. II.56.

Controlul predictiv cu restricții probabilistice a fost proiectat folosind soluția Problemei II.2.9, având următorii parametri: $N = 2$, $Q = 10000I_3$, $R = 2$, $S_{x0} = I_3$ și $S_W = I_3$.

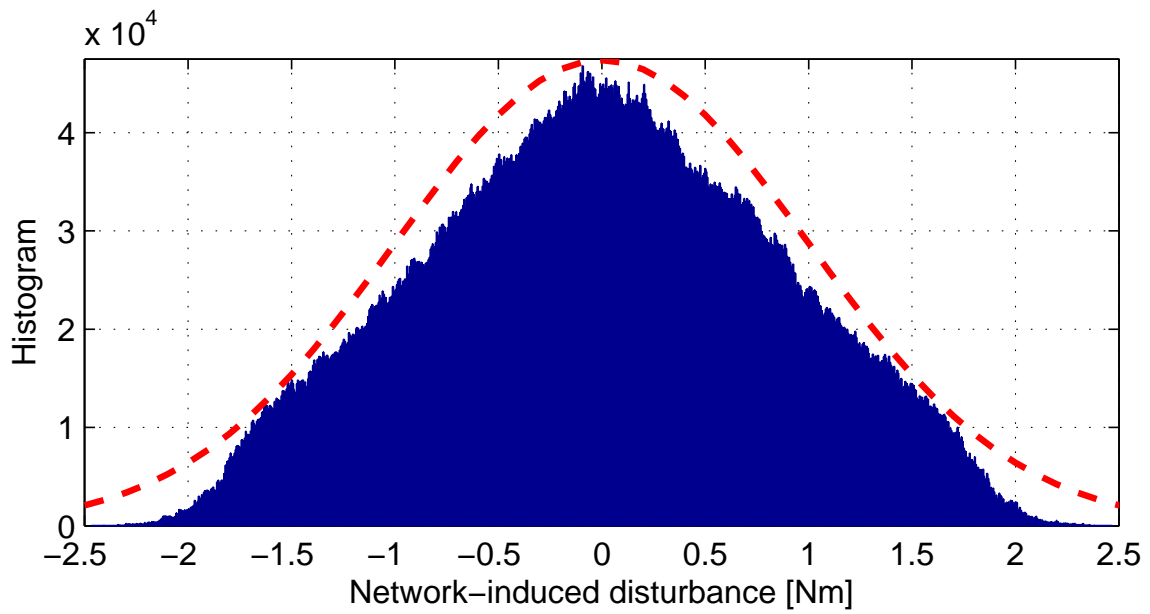


Figura II.55: Histograma bazata pe efectul întârzierilor. Linia punctată reprezintă distribuția gaussiană .

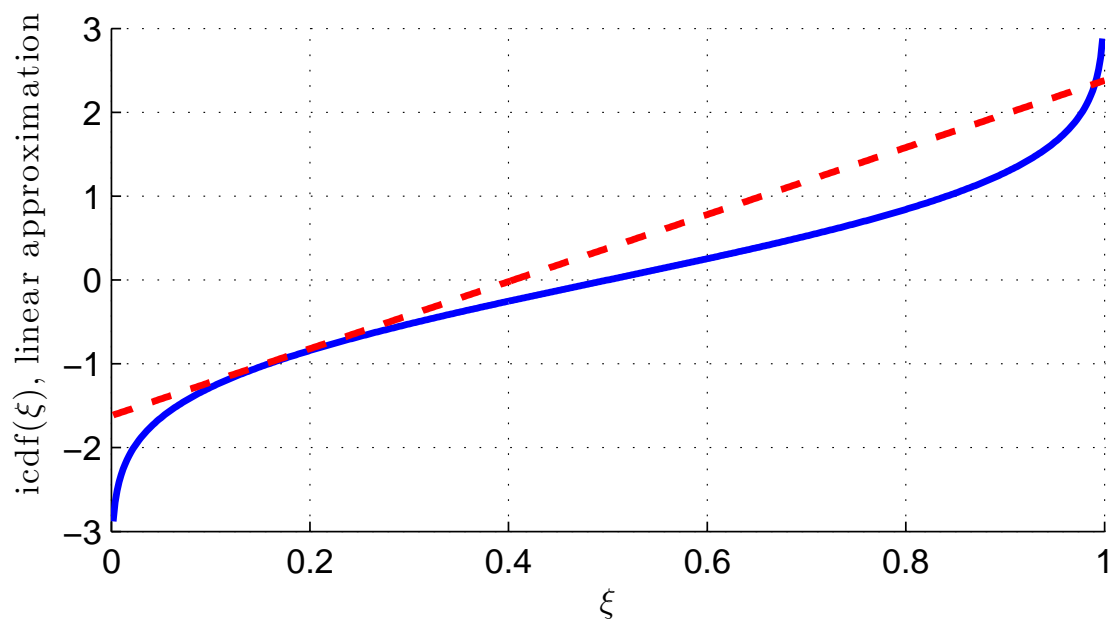


Figura II.56: Funcția de distribuție cumulativă inversă și aproximarea liniară (linia punctată).

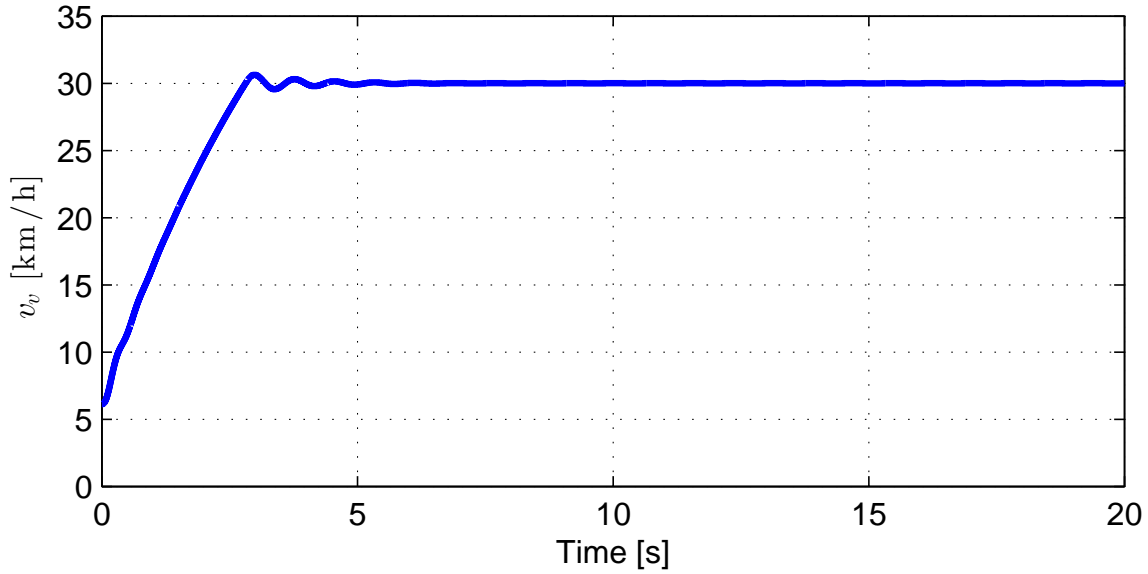


Figura II.57: Viteza vehiculului.

Limitele pentru restricțiile din (II.164) și (II.165) au fost alese ca $T_e^{\max} = 120$ Nm, $\omega_e^{\min} = 600$ rpm și $\omega_e^{\max} = 5000$ rpm. Viteza unghiulară de referință a fost fixată la $v_v^r = x_2^{ss} \cdot 3.6 \cdot r_w = 30$ km/h.

Problema II.2.9 a fost evaluată pentru valori diferite ale vitezei de torsiune maxime x_3^{\max} în (II.166), pentru a analiza influența acestora asupra probabilității minime δ care poate fi impusă. Într-adevăr, câte o sută de experimente au fost derulate pentru fiecare valoare x_3^{\max} pentru a verifica efectul întârzierilor stohastice asupra performanței sistemului de control. Rezultatele sunt ilustrate în Figurile II.57-II.59. În Fig. II.57 este prezentată viteza vehiculului, și se poate observa că performanțele de reglare sunt foarte bune. Viteza de torsiune poate fi văzută în Fig. II.58; se poate observa că valoarea de regim staționar pentru x_3 este 0.006 rad. Cuplul motorului este reprezentat în Fig. II.59, unde se poate observa că limita superioară definită de restricția asupra comenzii este satisfăcută.

Datele ilustrate în Fig. II.60 arată că restricțiile de probabilitate δ sunt satisfăcute de valoarea maximă setată pentru viteza de torsiune. Se poate observa că probabilitatea ca viteza de torsiune x_3 să depășească 2.4 rad este 0, în timp ce probabilitatea ca x_3 să depășească 0.007 rad se apropie de 0.6 (a se observa că valoarea de regim staționar pentru x_3 este 0.006 rad).

II.2.9 Control predictiv robust pentru sistemele controlate în rețea considerând pierderile de pachete de date ca perturbații

Factorul cheie în determinarea calității aplicațiilor care sunt sensibile la întârzieri, cum ar fi aplicațiile din domeniul auto critice din punct de vedere al timpului, este reprezentat de pierderea pachetelor de date. Aceste aplicații se confruntă cu degradarea calității și a performanțelor în timp ce cresc pierderile de pachete de date și / sau întârzieri în rețeaua de comunicații. Practica sugerează corelarea pierderilor de pachete de date. În plus, apariția

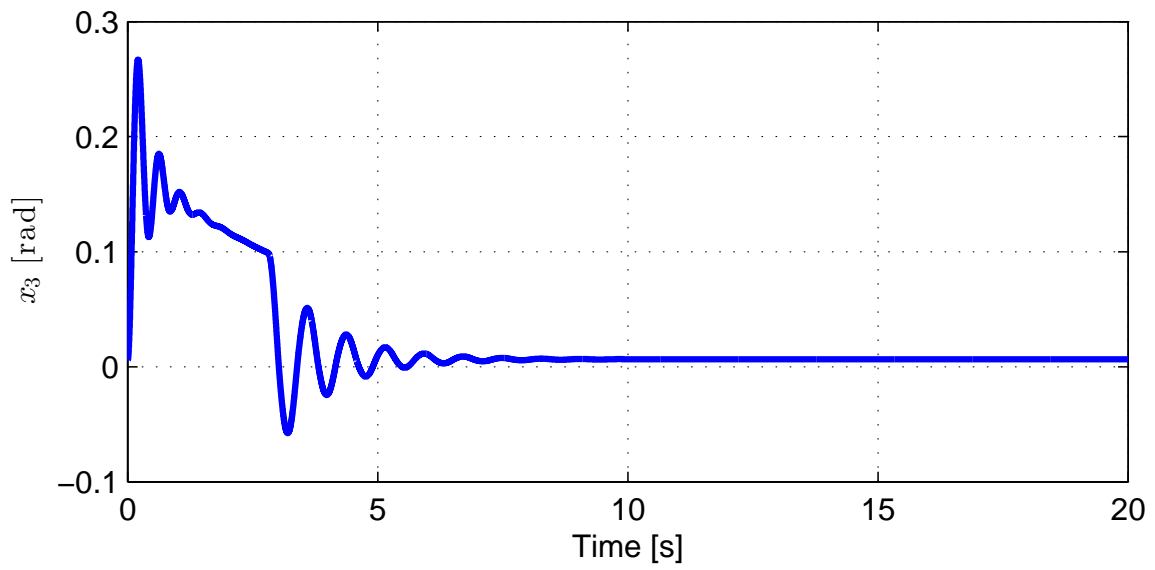


Figura II.58: Viteza de torsiune.

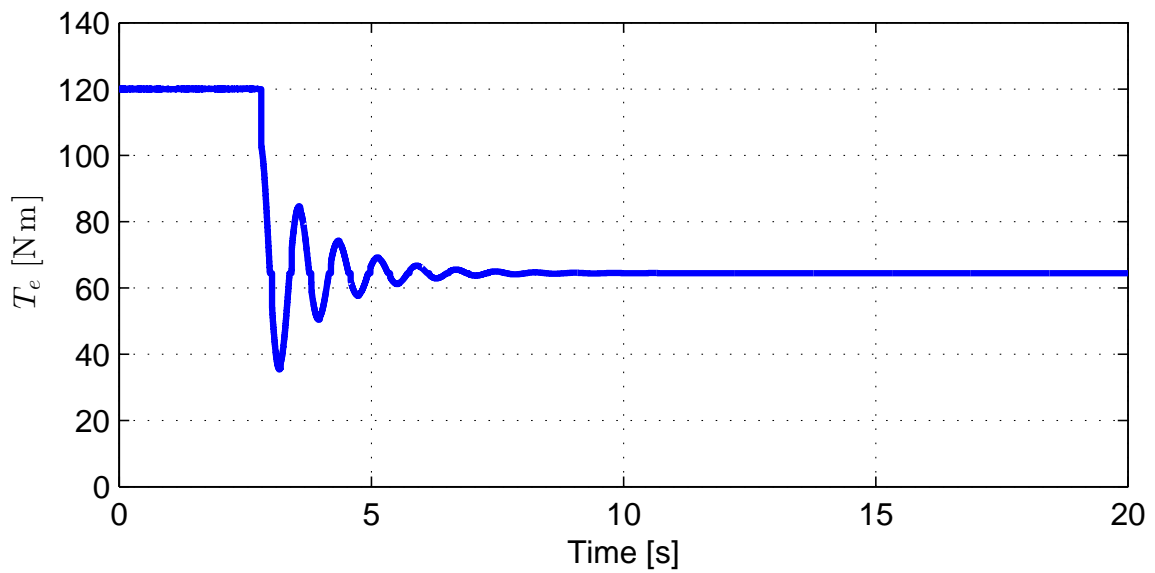


Figura II.59: Cuplul motorului (semnalul de control).

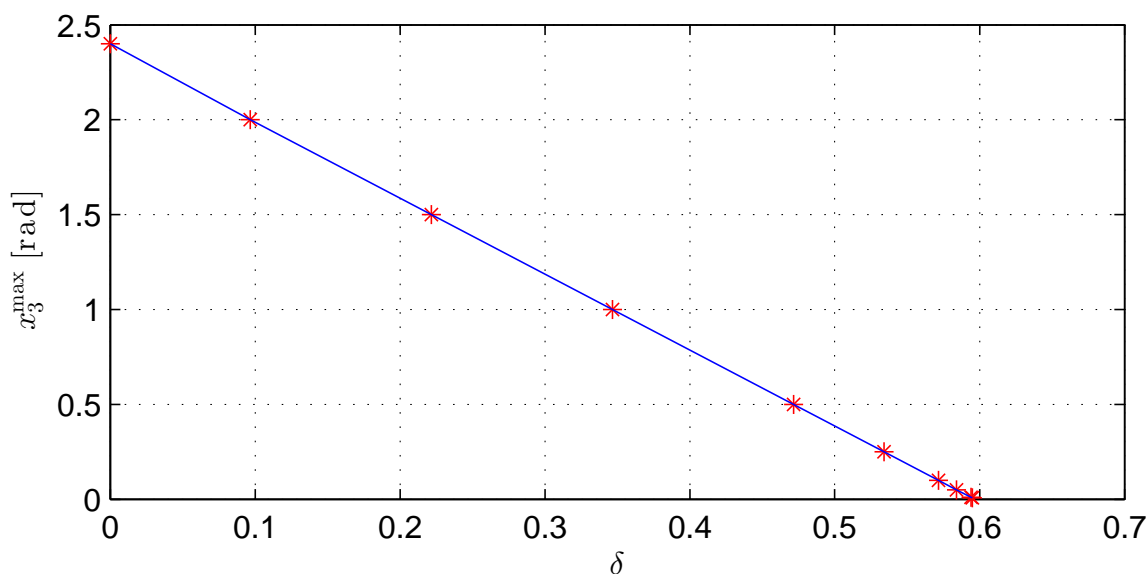


Figura II.60: Caracteristica între probabilitate și valoarea maximă a vitezei de torsiune.

lor este, de obicei, înșiruită, ceea ce înseamnă că următorul pachet după o pierdere este cel mai probabil, de asemenea, o pierdere. Pierderile de pachete de date sunt cauzate în principal de defectele receptorului și de condițiile canalului, de exemplu degradarea calității legăturii și congestia canalului. O altă explicație pentru apariția pierderilor de pachete înșiruite provine din faptul că rata de transmisie a pachetelor este de obicei mai mare decât ritmul la care condițiile de linie de transmisie se schimbă de obicei.

În [238] și [239], autorii consideră că pierderea pachetelor de date este doar pe calea directă, în timp ce în lucrările [240–243] sunt considerate pierderile pe calea de reacție.

În această secțiune se propune o metodologie de modelare a pierderilor de pachete de date induse de rețea atât pe calea directă, cât și pe cea de reacție ca perturbații. Pentru a include perturbațiile introduse de rețea, se arată că poate fi utilizată limitarea semnalului de control. Astfel, în proiectarea controlului predictiv, trebuie luată în considerare aceasta legătură. Algoritmul de control predictiv este dezvoltat într-un cadru robust, utilizând funcțiile flexibile de control Lyapunov (FCLF) pentru asigurarea stabilității pe stare. Metoda de modelare a pierderilor de pachete de date ca perturbații este o adaptare a metodei propuse în [35, 66, 244], unde autorii propun o metodă de modelare a întârzierilor datelor pe bază de perturbații în sistemele de control în rețea.

Rezultatele teoretice sunt ilustrate printr-un exemplu al industriei de automobile care utilizează controlul unui sistem de direcție asistată electric (EPAS) prin intermediul rețelei CAN. Controlul EPAS este dificil, deoarece trebuie să se rezolve mai multe sarcini, cum ar fi generarea de cuplu de asistență, amortizarea vibrațiilor, furnizarea unui feedback corect al direcției pentru șofer și robustețea de perturbare (posibil cauzată de pierderea pachetelor de date în rețea). Regulatorul are nevoie de măsurători ale cuplului aplicat de șofer, ale unghiului volanului și vitezei motorului electric. Toate aceste obiective luate separat ar putea fi gestionate prin utilizarea a diferite bucle închise de reglare pentru fiecare subsistem. Pentru a elimina dezavantajele utilizării mai multor bucle de reglare și acordarea individuală

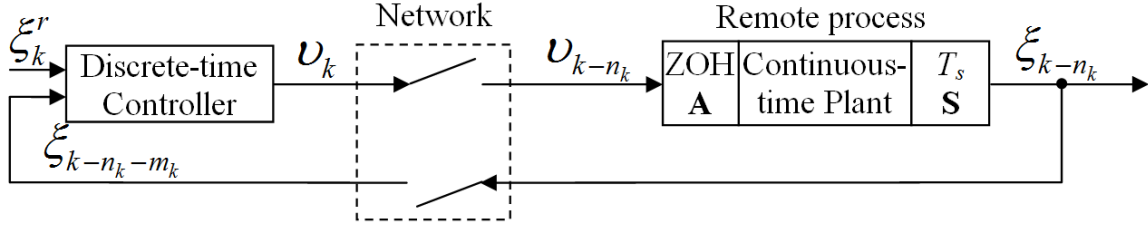


Figura II.61: Pierderi de date în structura de reglare în rețea.

a fiecărui regulator, în această secțiune este propus un regulator predictiv. Au fost efectuate simulări în Matlab / Simulink, iar rezultatele obținute arată că metodologia de modelare și strategia de control sunt eficiente în obținerea performanțelor dorite.

II.2.9.1 Modelarea pierderilor de pachete de date ca perturbații

Această secțiune introduce un cadru general pentru modelarea pierderilor de date introduse de rețeaua de comunicații sub formă de perturbații. Mai mult, regulatorul predictiv este proiectat utilizând modelul prezentat.

Fie n_k și m_k mărimea pierderii consecutive de pachete de date pentru semnalul de comandă, respectiv pentru semnalul de măsură, la momentul discret de timp $k \in \mathbb{Z}_+$. Mai mult decât atât, fie \bar{n} și \bar{m} dimensiunea maximă a pierderii consecutive de pachete de date pentru semnalul de comandă și respectiv pentru semnalul de măsură, adică $n_k \in \mathbb{Z}_{[0, \bar{n}]}$, $m_k \in \mathbb{Z}_{[0, \bar{m}]}$. În timpul pierderilor de pachete de date consecutive, ultimul pachet recepționat rămâne activ până la sosirea unui nou pachet.

În continuare, este luată în considerare o structură de control în rețea standard, așa cum este ilustrat în Fig. II.61. Arhitectura sistemului de control este construită în jurul unei rețele de comunicații, în care apar pierderi de pachete de date și care este alcătuită dintr-un proces fizic continuu cu un nod element de execuție (A) și nod senzor (S) asociate și un nod al regulatorului discret.

Modelul sistemului de control în rețea Modelul continuu al sistemului cu pierderi de date pe intrare și ieșire se poate formula după cum urmează

$$\begin{aligned} \dot{\xi}(t) &= A_c \xi(t) + b_c v^*(t), \\ v^*(t) &= v_k, \forall t \in [t_k + n_k, t_{k+1} + n_{k+1}), \end{aligned} \quad (\text{II.171})$$

unde $t_k = kT_s, k \in \mathbb{Z}_+$ presupunând că $v^*(t) = v_0 \forall t \in [0, n_0]$ cu $v_0 \in \mathbb{R}$ un vector constant calculat apriori, și cu n_0 numărul inițial de pierderi de pachete de date. Legea de control obținută la momentul de timp $t = t_k$ este $v_k \in \mathbb{R}$.

Reamintim că dimensiunea maximă a pierderilor de pachete consecutive a semnalului de comandă este notată cu \bar{n} , unde $\bar{n} \in \mathbb{Z}_+$. Presupunând $v_k = \psi_k \forall k \in \mathbb{Z}_{[-\bar{n}-1, -1]}$ cu $\psi_{[-\bar{n}-1, -1]}$ un vector calculat apriori, modelul discret rezultă

$$\begin{aligned} \xi_{k+1} &= A_d \xi_k + b_d v_k + \Delta_0(n_k)(v_{k-1} - v_k) \\ &\quad + \Delta_1(n_k)(v_{k-2} - v_{k-1}) + \dots \\ &\quad + \Delta_{\bar{n}}(n_k)(v_{k-\bar{n}-1} - v_{k-\bar{n}}), \end{aligned} \quad (\text{II.172})$$

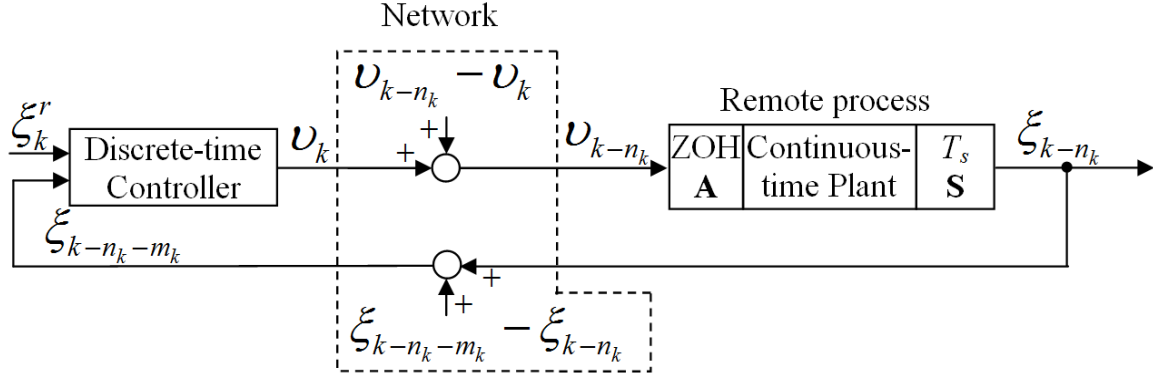


Figura II.62: Sistem de control în rețea cu pierderi de date modelate ca perturbații.

unde $A_d = e^{A_c T_s}$, $b_d = \int_0^{T_s} e^{A_c(T_s-\theta)} d\theta b_c$,

$$\Delta_i(n_k) := \begin{cases} 0, & n_{k-i} - i \leq 0 \\ \int_0^{T_s} e^{A_c(T_s-\theta)} d\theta b_c, & 1 \leq n_{k-i} - i \end{cases} \quad (\text{II.173})$$

pentru $k \in \mathbb{Z}_+$ and $i \in \mathbb{Z}_{[0, \bar{n}]}$.

$$\xi_{k+1} = A_d \xi_k + b_d v_k + \sum_{i=0}^{\bar{n}} \Delta_i(v_{k-i-1} - v_{k-i}). \quad (\text{II.174})$$

Trebuie remarcat faptul că vectorul de intrare $v_{[k-\bar{n}-1, k]}$ este cunoscut la momentul de timp $k \in \mathbb{Z}_+$.

Pierderile de pachete de date influențează semnalele de intrare și de ieșire ale sistemului, așa cum este ilustrat în Fig. II.61. Considerând pierderile pachetelor de date ca perturbații, structura de control în buclă închisă prezentată în Fig. 1 este echivalentă cu cea ilustrată în Fig. 2, adică perturbația pe calea directă este diferența dintre semnalul de comandă recepționat de servomotor și cea transmisă de către regulator, iar perturbația căii de reacție este diferența dintre ieșirea primită de către regulator și utilizată pentru calculul mărimii de comandă și ieșirea efectivă a sistemului reglat.

Pierderi de date pe calea directă Perturbația pe calea directă este reprezentată în timp discret ca

$$v_k^d = v_{k-n_k} - v_k. \quad (\text{II.175})$$

În majoritatea aplicațiilor de inginerie în timp real, limitarea intrării procesului este obligatorie. Luând în considerare acest lucru, trebuie să găsim un set Ω_v , care este limitat și include toate posibilele perturbații care apar pe calea directă datorită pierderilor de pachete de date.

Considerând explicit limitele semnalului de control ca

$$v^m \leq v_k \leq v^M, \quad (\text{II.176})$$

unde limitele minimă v^m și maximă v^M sunt nenegative. Ca urmare, perturbațiile induse de rețeaua de comunicații pot fi limitate astfel

$$v^m - v^M \leq v_k^d \leq v^M - v^m. \quad (\text{II.177})$$

Mai mult decât atât, dacă mai presupunem o restricție incrementală asupra legii de reglare discrete

$$-v^\Delta \leq \Delta v_k \leq v^\Delta, \quad (\text{II.178})$$

unde $\Delta v_k := v_k - v_{k-1}$, $\forall k \in \mathbb{Z}_{\geq 1}$, cu v_0 o valoare calculata anterior si v^Δ este rata de schimbare maxima a semnalului de control la fiecare moment de esantionare $k \in \mathbb{Z}_{\geq 1}$, perturbația semnalului de control poate fi limitată ca

$$-\bar{n}v^\Delta \leq v_k^d \leq \bar{n}v^\Delta. \quad (\text{II.179})$$

Pierderi de date pe calea de reacție Semnalul de reacție (măsurătoarea trimisă de senzor către regulator) este influențată de perturbațiile ce apar pe ieșirea sistemului reglat, aceste perturbații fiind produse de pierderile de date de pe linia de comunicații a căii de reacție. Deci, perturbația care influențează linia de reacție se poate scrie discret ca

$$\xi_k^d = \xi_{k-n_k-m_k} - \xi_{k-n_k}. \quad (\text{II.180})$$

Luând în considerare limitele intrării procesului la distanță, ar trebui găsit un set limitat Ω_ξ care să includă toate perturbațiile posibile care pot apărea pe linia de reacție datorită pierderilor de date.

Se consideră modelul discret al părții fixate cu perturbații pe intrare

$$\begin{aligned} \xi_{k+1} &= A_d \xi_k + b_d(v_k + v_k^d) = \\ &= A_d^2 \xi_{k-1} + A_d b_d(v_{k-1} + v_{k-1}^d) + b_d(v_k + v_k^d) = \\ &= A_d^j \xi_{k-j+1} + A_d^{j-1} b_d(v_{k-j+1} + v_{k-j+1}^d) + \dots \\ &\quad \dots + A_d b_d(v_{k-1} + v_{k-1}^d) + b_d(v_k + v_k^d) = \\ &= A_d^j \xi_{k-j+1} + \sum_{i=0}^{j-1} A_d^i b_d(v_{k-i} + v_{k-i}^d). \end{aligned} \quad (\text{II.181})$$

Calculând

$$\begin{aligned} \xi_{k-n_k} &= A_d \xi_{k-n_k-1} + b_d(v_{k-n_k-1} + v_{k-n_k-1}^d) = \\ &= A_d^{m_k} \xi_{k-n_k-m_k} + \\ &\quad + A_d^{m_k-1} b_d(v_{k-n_k-m_k} + v_{k-n_k-m_k}^d) + \dots \\ &\quad \dots + A_d b_d(v_{k-n_k-2} + v_{k-n_k-2}^d) + \\ &\quad + b_d(v_{k-n_k-1} + v_{k-n_k-1}^d) = \\ &= A_d^{m_k} \xi_{k-n_k-m_k} + \\ &\quad + \sum_{i=0}^{m_k-1} A_d^i b_d(v_{k-n_k-i-1} + v_{k-n_k-i-1}^d) \end{aligned}$$

(II.180) devine

$$\begin{aligned} \xi_k^d &= (I_n - A_d^{m_k}) \xi_{k-n_k-m_k} - \\ &\quad - \sum_{i=0}^{m_k-1} A_d^i b_d (v_{k-n_k-i-1} + v_{k-n_k-i-1}^d) \end{aligned} \quad (\text{II.182})$$

Se poate observa că valorile $\xi_{k-n_k-m_k}$ sunt cunoscute la momentele de timp discrete k , toate valorile de control v_{k-i} , $i \in \mathbb{Z}_{\geq 1}$ sunt cunoscute, toate perturbațiile de pe calea directă v_{k-i}^d , $i \in \mathbb{Z}_{\geq 1}$ sunt mărginite, n_k și m_k sunt mărginite de \bar{n} și \bar{m} , respectiv, astfel încât ξ_k^d poate fi mărginit la fiecare moment de timp discret $k \in \mathbb{Z}_+$.

Din punctul de vedere al regulatorului, considerând cele prezentate mai sus, modelul complet al procesului rezultă

$$\xi_{k+1} = A_d \xi_k + b_d v_k + b_d v_k^d + \xi_k^d. \quad (\text{II.183})$$

Dupa cum a fost aratat, pierderile de date produc o perturbație variabilă în timp, dar seturile

$$\begin{aligned} \Omega_v &= \{b_d v_k^d \in \mathbb{R}^n \mid -\bar{a}v^\Delta \leq v_k^d \leq \bar{a}v^\Delta\}, \\ \Omega_\xi &= \{\xi_k^d \in \mathbb{R}^n \mid \xi_k^d \text{ satisfies relation (II.182)}\}, \end{aligned} \quad (\text{II.184})$$

rămân fixe, deoarece limitele calculate anterior pot substitui aceste perturbații. Metoda de modelare propusă poate fi folosită în contextul propus în [68], unde perturbațiile sunt luate în considerare încă din faza de proiectare a regulatorului predictiv. Aceasta va fi arată în secțiunea următoare.

II.2.9.2 Metodologia reglării predictive robuste

În [245] condiția de stabilitate Lyapunov convențională a fost extinsă, iar noul concept a fost denumit *funcții Lyapunov flexibile*. Aceste funcții Lyapunov pot garanta stabilitatea unui sistem de control predictiv (presupunând că fezabilitatea recursivă este satisfăcută), chiar dacă orizontul de predicție ales este scurt. Prin urmare, această abordare va fi folosită în cele ce urmează, obținând astfel un regulator predictiv robust cu un orizont de predicție 1, care folosește funcții Lyapunov flexibile.

Să considerăm sistemul discret neliniar

$$\xi_{k+1} = A_d \xi_k + b_d v_k + \omega_k, \quad k \in \mathbb{Z}_+, \quad (\text{II.185})$$

unde $\xi_k \in \Xi \subseteq \mathbb{R}^n$ este vectorul stărilor, $v_k \in \Upsilon \subseteq \mathbb{R}^m$ este vectorul intrărilor și $\omega_k \in \Omega \subseteq \mathbb{R}^n$ este o perturbație cunoscută la momentul de eșantionare k . În ceea ce urmează, să presupunem că seturile Ξ (set de stări fezabile), Υ (set de intrări fezabile) și Ω (set de perturbații) sunt poliedre mărginite nenule. Fie $\alpha_1, \alpha_2, \alpha_3 \in \mathcal{K}_\infty$ și $\sigma \in \mathcal{K}$.

Definiția II.2.16 Considerând o funcție $V : \mathbb{R}^n \rightarrow \mathbb{R}_+$ pentru care următoarele inegalități sunt îndeplinite

$$\alpha_1(\|\xi\|) \leq V(\xi) \leq \alpha_2(\|\xi\|), \quad \forall \xi \in \mathbb{R}^n. \quad (\text{II.186})$$

Pentru funcția V există o lege de control $\pi : \mathbb{R}^n \rightrightarrows \Upsilon$ care satisface

$$\begin{aligned} V(A_d \xi_k + b_d v_k + w_k) - V(\xi) &\leq \\ &\leq -\alpha_3(\|\xi\|) + \sigma(\|\omega\|), \\ &\forall \xi \in \Xi, \forall v \in \pi(\xi), \forall \omega \in \Omega. \end{aligned} \quad (\text{II.187})$$

Funcția V se numește *funcție de control Lyapunov de stabilitate intrare-stare* din Ξ pentru sistemul (III.36) și perturbațiile din Ω .

Pentru a proiecta regulatorul predictiv cu stabilitate intrare-stare asigurând rejectarea perturbației, așa cum este făcut în [68] poate fi folosită teoria stabilității intrare-stare [246].

Fie Ω un corp convex cu vârfurile $\omega^e, e = 1, \dots, E$, și fie $\lambda_k^e, k \in \mathbb{Z}_+$ variabilele după care se face optimizarea pentru fiecare din vârfuri ω^e . Fie $J(\lambda^1, \dots, \lambda^E, \lambda) : \mathbb{R}_+^E \times \mathbb{R}_+ \rightarrow \mathbb{R}_+$ o funcție strict convexă, nemărginită radial ($J(\cdot)$ tinde la infinit atunci când argumentele tind la infinit) și fie $J(\lambda^1, \dots, \lambda^E, \lambda) \rightarrow 0 \Rightarrow \lambda^e \rightarrow 0$, pentru $e = 1, \dots, E$ și $\lambda \rightarrow 0$ și $J(0, \dots, 0, 0) = 0$.

Se alege o funcție Lyapunov $V(\cdot)$ apriori pentru sistemul (III.36), fără perturbații, și fie $\alpha_3 \in \mathcal{K}_\infty$ și $\xi \in \Xi$ dați. La fiecare moment de eșantionare $k \in \mathbb{Z}_+$, algoritmul de control predictiv propus va rezolva următoarea problemă

Problema II.2.17 La momentul discret de timp $k \in \mathbb{Z}_+$ se măsoara starea ξ_k și se minimizează funcția de cost $J(\lambda_k^1, \dots, \lambda_k^E, \lambda_k)$ prin $v_k, \lambda_k^1, \dots, \lambda_k^E$ și λ_k , care sunt supuse restricțiilor

$$v_k \in \mathbb{U}, (A_d \xi_k + b_d v_k) \in \Xi, \lambda_k^e \geq 0, \lambda_k \geq 0, \quad (\text{II.188a})$$

$$V(A_d \xi_k + b_d v_k) - V(\xi_k) + \alpha_3(\|\xi_k\|) \leq \lambda_k, \quad (\text{II.188b})$$

$$V(A_d \xi_k + b_d v_k + \omega^e) - V(\xi_k) + \alpha_3(\|\xi_k\|) \leq \lambda_k^e, \quad (\text{II.188c})$$

$\forall e = 1, \dots, E$. □

Fie $\pi(\xi_k) := \{v_k \in \mathbb{R}^m \mid \exists \lambda_k, \lambda_k^e, e \in \mathbb{Z}_{[1,E]} \text{ s.t. (II.188) holds}\}$ și fie $\phi_{cl}(\xi_k, \pi(\xi_k), \omega_k) := \{A_d \xi_k + b_d v_k + w_k \mid v_k \in \pi(\xi_k)\}$ incluziunea cu diferențe corespunzătoare sistemului (III.36) în buclă închisă folosind soluția fezabilă obținută la Problema II.2.17 la fiecare moment de eșantionare $k \in \mathbb{Z}_+$.

Rezultatele teoretice sunt prezentate în cele ce urmează. A fost folosită o adaptare a rezultatelor din [68] pentru a se potrivi relaxării (II.188b) din Problema II.2.17, adică $\lambda_k = 0 \forall k \in \mathbb{Z}_+$ corespunde problemei rezolvate în [68].

Teorema II.2.18 Fie $\alpha_1, \alpha_2, \alpha_3 \in \mathcal{K}_\infty$, o funcție continuă și convexă Lyapunov $V(\cdot)$ și funcția de cost $J(\cdot)$ date. Presupunând că Problema II.2.17 este fezabilă pentru toate stările ξ din Ξ și presupunând că $\lim_{k \rightarrow \infty} \lambda_k^* = 0$, atunci traiectoriile generate de

$$\xi_{k+1} \in \phi_{cl}(\xi_k, \pi(\xi_k), \omega_k), \quad k \in \mathbb{Z}_+, \quad (\text{II.189})$$

cu stările inițiale $\xi_0 \in \Xi$ converg în timp finit la un set robust pozitiv semidefinit al Ξ , în care stabilitatea intrare-ieșire este garantată în Ω .

Teorema II.2.18 poate fi demonstrată folosind argumentații standard ca și în cazul demonstrării stabilității Lyapunov și stabilității intrare-stare. Cititorii interesați pot găsi mai multe detalii în [68] și [245]. Deci, regulatorul propus are avantajul că este stabil intrare-stare, oricare ar fi o soluție fezabilă a optimizării. Condiția de limitare $\lim_{k \rightarrow \infty} \lambda_k^* = 0$ este esențială în demonstrarea stabilității, deci, pentru a garanta această condiție, în ceea ce urmează se prezintă o soluție ne-conservativă.

Lema II.2.19 Considerând o constantă cunoscută a priori $\delta \in \mathbb{R}_+$ și folosind $\rho \in \mathbb{R}_{(0,1)}$ și $\mu \in \mathbb{Z}_{>0}$. Dacă

$$0 \leq \lambda_k \leq \rho^{\frac{1}{\mu}} (\lambda_{k-1}^* + \rho^{\frac{k-1}{\mu}} \delta), \quad \forall k \in \mathbb{Z}_{\geq 1}, \quad (\text{II.190})$$

atunci $\lim_{k \rightarrow \infty} \lambda_k = 0$.

Demonstrația Lemei II.2.19 poate fi găsită în [247], care spune ca stabilitatea asimptotică este asigurată prin îndeplinirea proprietății $\lim_{k \rightarrow \infty} \lambda_k^* = 0$, care este satisfăcută prin adăugarea restricției (II.190) la Problema II.2.17.

Mai mult decât atât, a fost arătat în [248] că folosind funcții Lyapunov bazate pe norma infinit, algoritmul de control predictiv propus poate fi implementat prin rezolvarea unei singure probleme de programare liniară la fiecare pas de eșantionare.

II.2.9.3 Modelul sistemului de servodirecție electric

În zilele noastre, în multe vehicule, sistemele de servodirecție hidraulică (HPS) sunt înlocuite cu sisteme de direcție asistate electric. În comparație cu sistemele HPS tradiționale, sistemele EPAS au multe avantaje: consum redus de combustibil, independența motorului, simțire realistă a direcției, asamblare rapidă, modularitate și dimensiune redusă [249].

Modelul dinamic al sistemului de direcție asistat electric Modelul dinamic al sistemului de servodirecție electric poate fi obținut prin studierea conexiunilor dintre motorul electric, mecanismul de direcție și forțele care apar datorită contactului cauciucurilor cu suprafața de rulare. Aplicând a doua lege a lui Newton se obține [250]

$$J_c \ddot{\theta}_c = -K_c \theta_c - B_c \dot{\theta}_c + K_c \frac{\theta_m}{N} - F_c \text{sign}(\dot{\theta}_c) + T_d, \quad (\text{II.191})$$

$$J_{eq} \ddot{\theta}_m = K_c \frac{\theta_c}{N} - \left(\frac{K_c}{N^2} + \frac{K_r R_p^2}{N^2} \right) \theta_m - B_{eq} \dot{\theta}_m + K_t I_m - F_m \text{sign}(\dot{\theta}_m) - \frac{R_p}{N} F_r, \quad (\text{II.192})$$

unde J_c este inerția coloanei de direcție, θ_c este unghiul volanului, K_c este rigiditatea coloanei de direcție, B_c este frecarea vâscoasă a coloanei de direcție, θ_m poziția unghiulară a motorului, N este reducția dintre motorul electric și coloană, F_c este frecarea coloanei de direcție, T_d este cuplul aplicat de șofer, $J_{eq} = J_m + \frac{R_p^2}{N^2} M_r$, cu J_m momentul de inerție al motorului, R_p este raza pinionului și M_r masa cremalierii; $B_{eq} = B_m + \frac{R_p^2}{N^2} B_r$, unde B_m este frecarea vâscoasă a motorului și B_r frecarea vâscoasă a cremalierii; K_r este elasticitatea cauciucului, K_t este constanta de cuplu a motorului, I_m este curentul prin motor,

F_m este frecarea motorului, F_r este forța de reacție a roții și $\theta_m = \frac{Nx_r}{R_p}$, cu x_r poziția cremalierii.

Motorul electric este descris de următoarele ecuații:

$$L\dot{I}_m = -RI_m - K_t\dot{\theta}_m + V, \quad (\text{II.193})$$

unde L este inductanța, R este rezistența și V este tensiunea aplicată motorului de curent continuu.

Modelul vehiculului Pentru a genera forța de reacție a drumului care acționează asupra cremalierii, se utilizează un model de vehicul. Modelul de vehicul ales este modelul bicicletă care modelează dinamica laterală a mașinii având următoarele stări: β - unghiul de alunecare al corpului, măsurat la centrul de greutate (CoG) și γ - viteza de rotație. Se presupune că există doar unghiuri mici și că forța laterală a cauciucului este proporțională cu unghiul de alunecare a cauciucului. Astfel, dinamica laterală a vehiculului este dată de următorul model liniar

$$\begin{aligned} \dot{\beta} &= -\frac{C_f + C_r}{mv_x}\beta + \left(-1 + \frac{l_r C_r + l_f C_f}{mv_x^2}\right)\gamma + \frac{C_f}{mv_x}\delta_f, \\ \dot{\gamma} &= \frac{l_r C_r + l_f C_f}{I_z}\beta - \frac{l_r^2 C_r + l_f^2 C_f}{I_z v_x}\gamma + \frac{l_f C_f}{I_z}\delta_f, \end{aligned} \quad (\text{II.194})$$

unde C_f și C_r sunt coeficienții de rigiditate la virare ale cauciucurilor de la roata din față și cea din spate, l_f și l_r sunt distanța de la centrul de greutate la puntea din față și puntea din spate, m este masa vehiculului, v_x este viteza longitudinală a vehiculului, $\delta_f = \frac{\theta_m}{N_v}$ este unghiul de virare al roții din față, cu N_v fiind reducerea dintre unghiul de virare al coloanei de direcție și unghiul roții, și I_z este momentul de inerție al vehiculului.

II.2.9.4 Rezultate de simulare

Metodologia de modelare a pierderii pachetelor de date propusă în secțiunea II.2.9.1 și strategia de control predictiv prezentată în secțiunea II.2.9.2 au fost testate în simulare pentru controlul sistemului EPAS. Obiectivul sistemului de control EPAS este generarea cuplului de asistență pentru șofer. Deoarece șoferul nu trebuie să simtă lipsa de asistență, face ca viteza de răspuns a sistemului de control să fie de o importanță critică.

Modelul EPAS dat de ecuațiile (II.191), (II.192) și (II.194) a fost liniarizat și discretizat cu o perioadă de eșantionare $T_s = 0.002$ s pentru proiectarea regulatorului.

Structura de control este similară cu cea utilizată în [250] cu o rețea de comunicații CAN care conectează regulatorul proiectat pentru sistemul de direcție, printr-o rețea de comunicații.

Din Fig. II.63, II.67 și II.68, se poate observa că pentru aceeași viteză a vehiculului (20m/s) (vezi Fig. II.69) unghiul de virare este mai mare decât atunci când nu se utilizează sistemul de asistență (Fig. II.69).

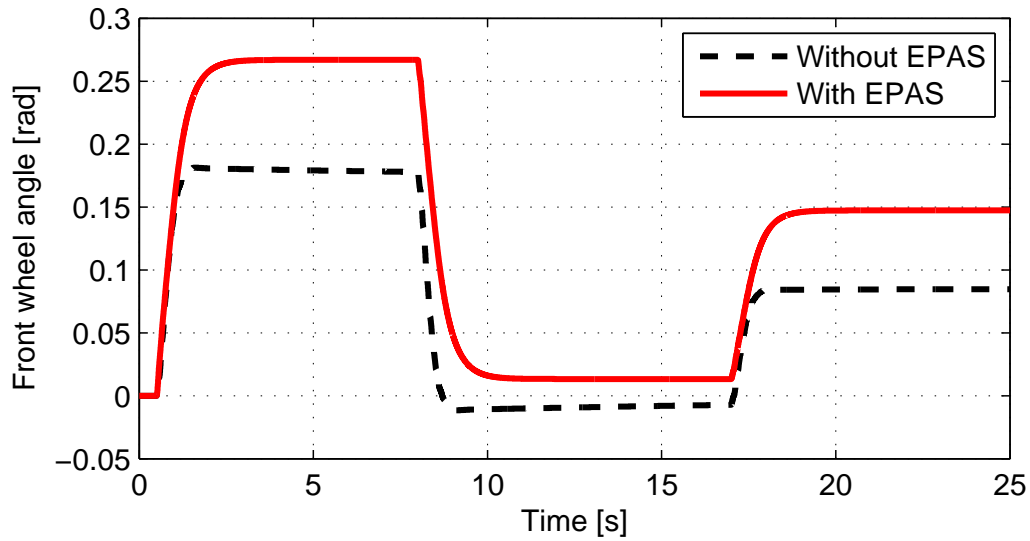


Figura II.63: Unghiul roții din față - α_f .

Pierderile de pachete de date care apar în rețeaua CAN pot fi observate în Fig. II.65 cu un detaliu în Fig. II.66, unde “1” înseamnă că pachetul este pierdut și “0” înseamnă că pachetul a ajuns la destinație.

Ieșirea sistemului EPAS poate fi observată în Fig. II.64, în care se poate observa că curentul motorului urmează curentul de referință. Tensiunea motorului, care este semnalul de comandă al sistemului, este ilustrată în figura II.70.

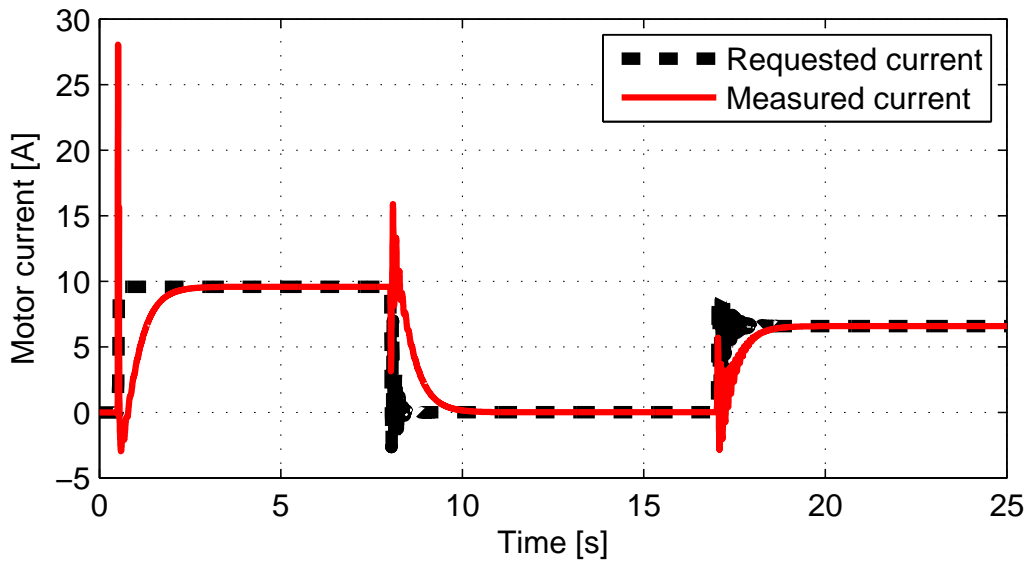


Figura II.64: Curentul - I_m .

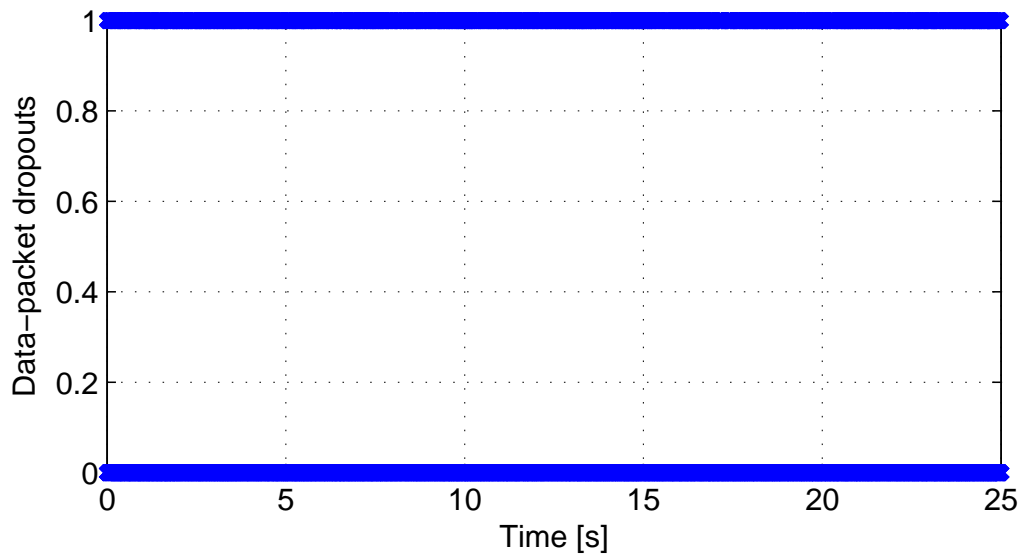


Figura II.65: Pierderi de pachete de date: "1" - pierderi, "0" - transmise cu succes.

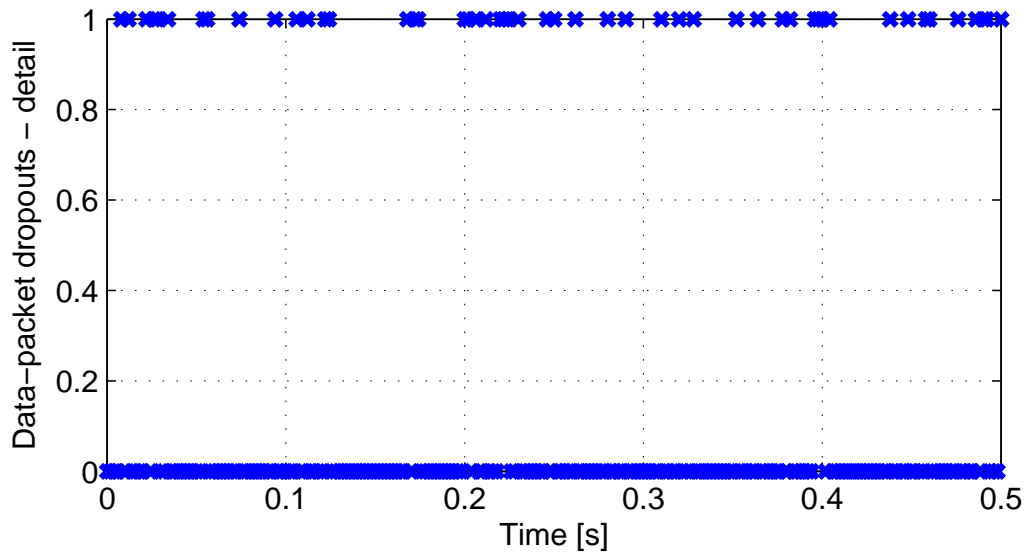


Figura II.66: Pierderi de pachete de date - detaliu: "1" - pierderi, "0" - transmise cu succes.

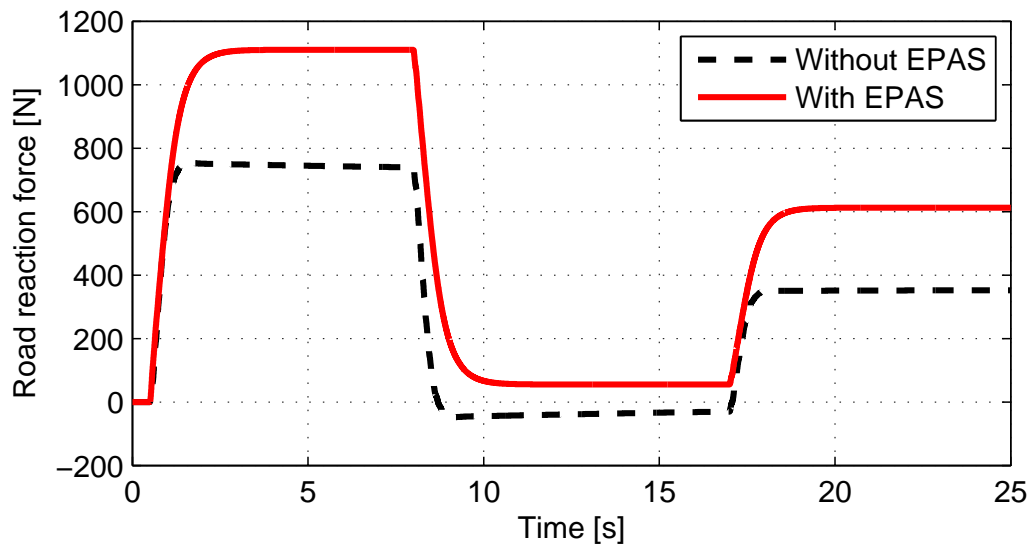


Figura II.67: Forța de reacție a suprafeței de rulare - F_r .

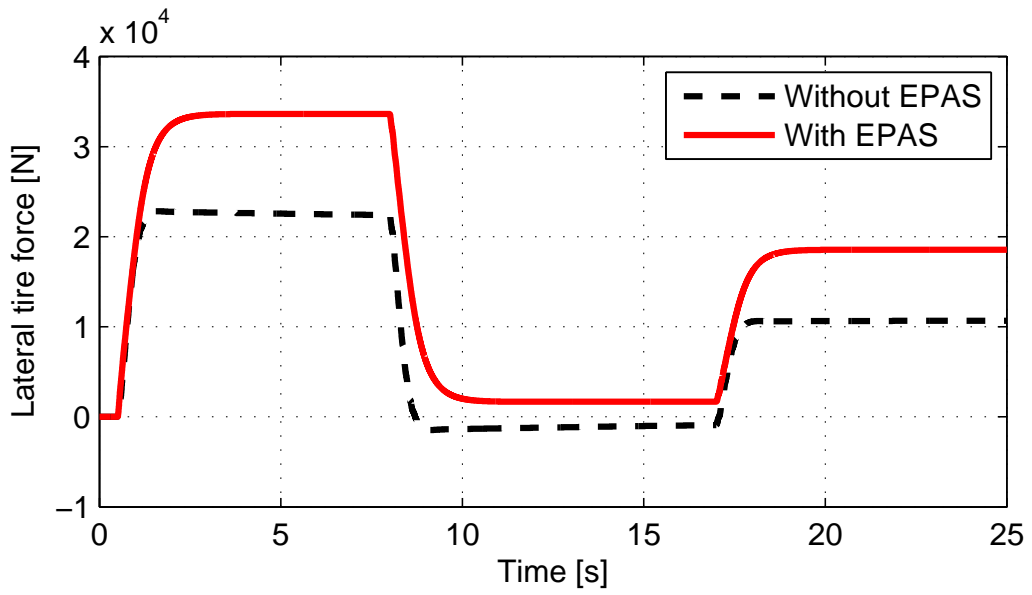


Figura II.68: Forța laterală a cauciucului - F_y .

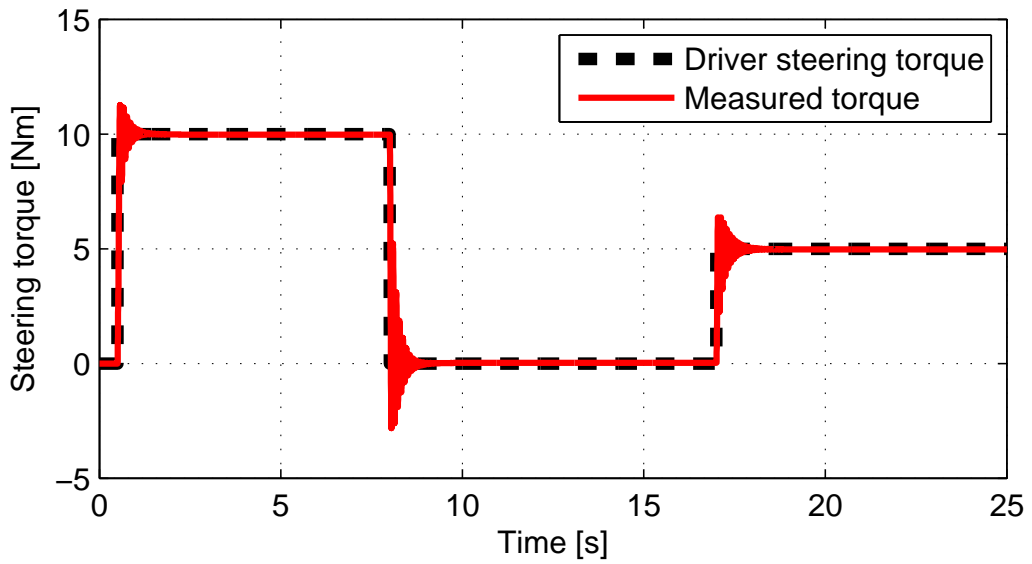


Figura II.69: Cuplul de virare - T_d .

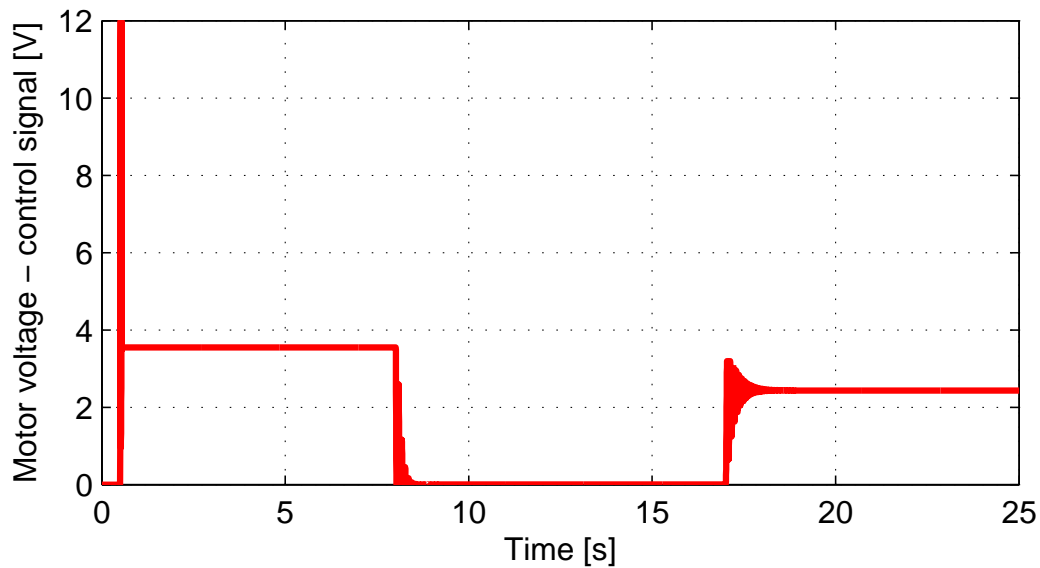


Figura II.70: tensiunea aplicată motorului (semnalul de control) - V .

II.3 (A2.2) Dezvoltarea de emulatoare în Matlab pentru plutoanele de vehicule cooperante

II.3.1 Emulator Matlab Simulink pentru un pluton de 5 vehicule modelate cu dublu integrator

În această secțiune se va descrie emulatorul Matlab Simulink folosit pentru simularea unui pluton de 5 vehicule prezentat în Figura II.71. Pentru o mai bună înțelegere a schemei, se va detalia separat în Figura II.72 ansamblul sub-sistem lider încadrat cu chenar roșu în timp ce ansamblul sub-sistem următor din chenarul albastru va fi prezentat în Figura II.73

Astfel, în Figura II.72 este reprezentat sub-sistemul lider, compus din două blocuri principale: *i*) funcția embeded care reprezintă agentul $i = 1$ care se ocupă cu rezolvarea funcției cost locală și *ii*) blocul Lti system care modelează procesul (în cazul curent, modelul augmentat (II.24)). Din punctul de vedere al agentului lider, funcția embeded are cinci intrări (i.e., cele două ieșiri poziția (poz) și viteza (vel) ale procesului, referințele impuse pentru poziție (ref_poz) respectiv viteză (ref_vel) și ceasul simulării ($clock$) care ține evidența timpului de simulare și trei ieșiri (i.e., comanda trimisă către proces ($Ut1$) care este primul element din vectorul optimal, traiectoria optimală sifatăă cu un element în variabilă incrementală ($deltaU_for_f1$) ce va fi folosită de catre primul vehiculul următor în problema de optimizare, și comanda optimală ($Ut1_for_f1$) identică cu ($Ut1$) ce se va folosi pentru a simula funcționarea modelului vehiculului următor).

Intrarea $clock$ este folosită doar pentru lider, pentru a predefini panta rampei referinței curente m din vectorul de referință predictat (II.26))

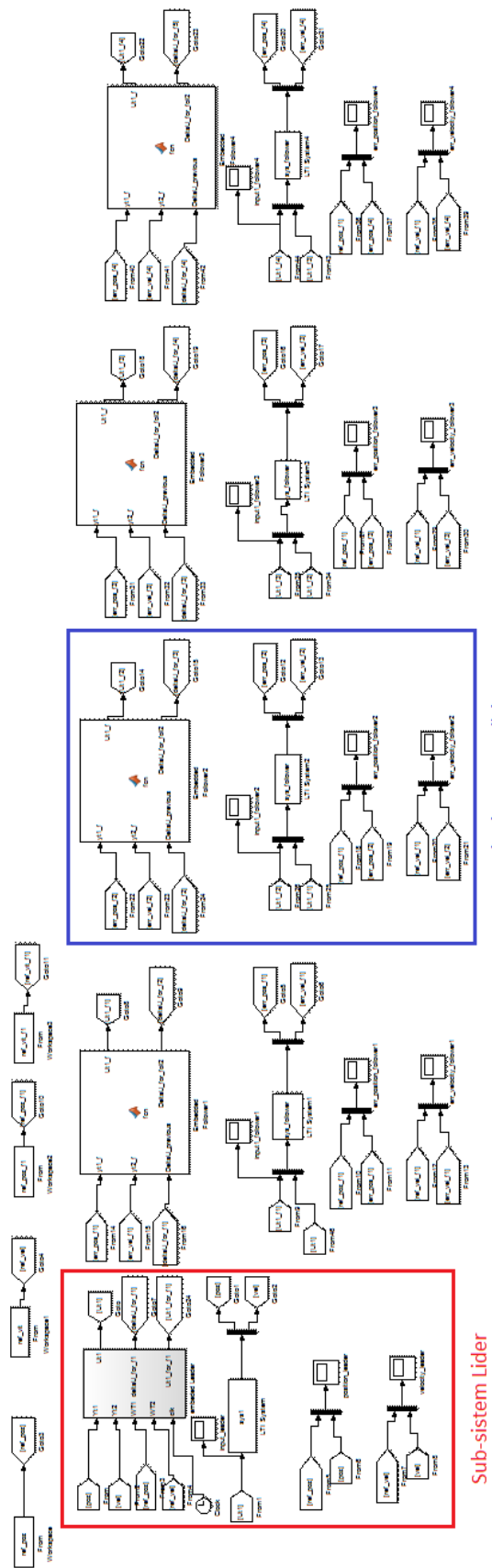


Figura II.71: Reprezentare grafică a emulatorului unui pluton de $N=5$ vehicule.

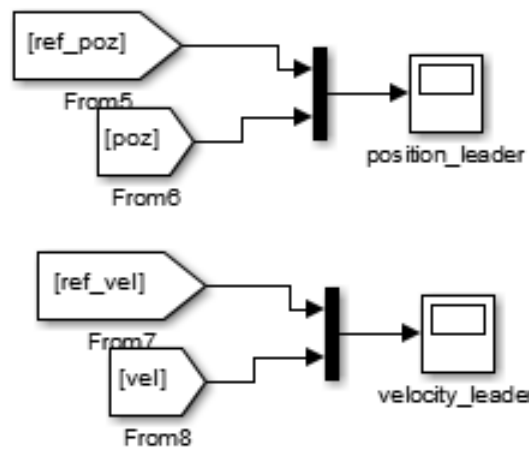
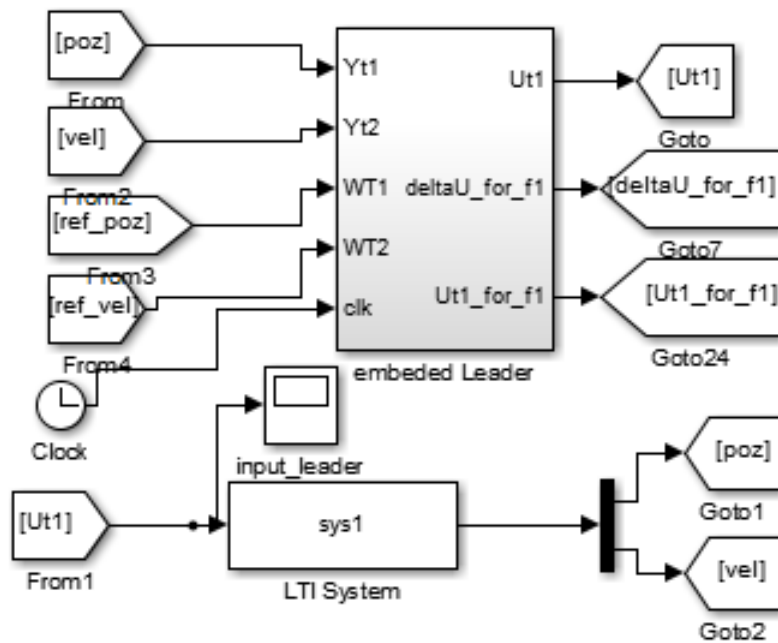


Figura II.72: Reprezentare grafică a ansamblului sub-sistem lider.

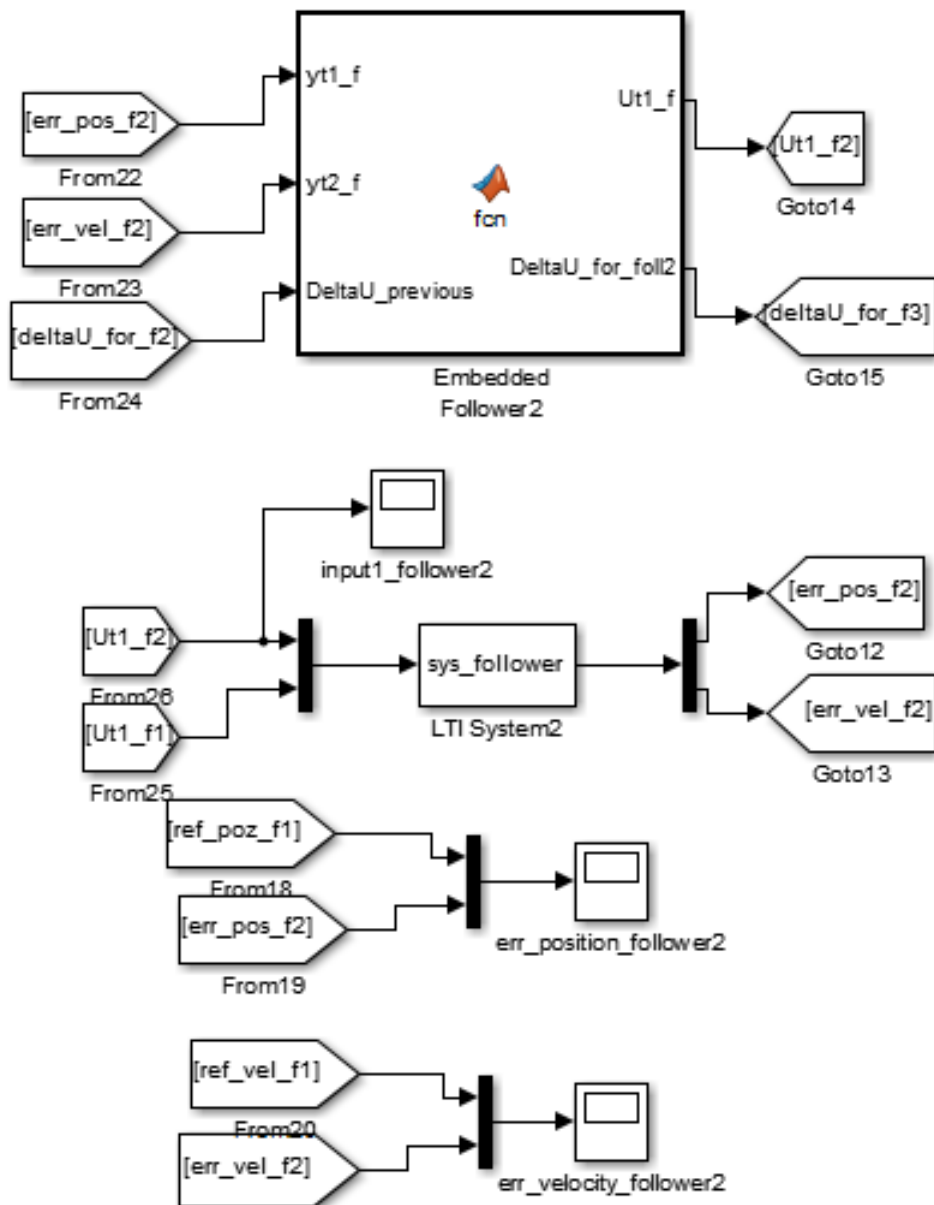


Figura II.73: Reprezentare grafică a ansamblului sub-sistem urmăritor.

Blocul care modelează sistemul, are ca mărime de intrare ieșirea agentului $Ut1$ și produce cele două ieșiri poz și vel care vor fi folosite la următorul eșantion de timp ca intrări în agent.

De asemenea se mai pot observa două blocuri de afișare a ieșirilor sistemului (i.e., *position_leader* care afișează poziția raportată la referința inpusă și *velocity_leader* unde se afișează viteza vehiculului).

De remarcat este faptul că ansamblele sub-sistem următor sunt identice din punct de vedere a funcționalității, ceea ce diferă fiind valorile numerice ale mărimilor de intrare respectiv ieșire, atât în blocul embeded agent cât și în sub-sistem.

Astfel, în Figura II.73 este reprezentat vehiculul #3 din pluton, adică al doilea vehicul următor din lanț. Agentul $i = 3$ primește la intrare eroarea de poziție (err_pos_f2), eroarea de viteză (err_vel_f2) și traiectoria optimală primită de la predecesor ($deltaU_for_f2$). Mărimile de ieșire sunt comanda trimisă către proces ($Ut1_f2$) care este primul element din vectorul optimal și traiectoria optimală siftată cu un element în variabilă incrementală ($deltaU_for_f3$) ce va fi folosită de către succesul său în problema de optimizare.

Blocul care simulează modelul sub-sistemului #3, are două intrări (primul element din vectorul optimal ($Ut1_f2$) calculat la fiecare perioadă de eșantionare conform principiului orizontului alunecător și primul element din vectorul optimal al vehiculului predecesor ($Ut1_f1$) calculat de către predecesor și trimis către succesul corespunzător). Ieșirile din sub-sistemul următor sunt erorile de poziție (err_pos_f2) și eroarea de viteză (err_vel_f2) ce vor fi folosite de către agent în problema de optimizare.

O caracteristică importantă a emulatorului propus este modularitatea propusă. După cum s-a menționat anterior, toate vehiculele următor au fost configurate identic, ceea ce permite extinderea nelimitată a numărului de vehicule în pluton fără a întâmpina mari dificultăți.

II.3.2 Emulator Matlab Simulink pentru un pluton de 5 vehicule modelate prin descrierea lanțului de transmisie a puterii

În această secțiune se va descrie emulatorul Matlab Simulink folosit pentru simularea unui pluton de 5 vehicule modelate prin descrierea lanțului de transmisie a puterii.

După cum s-a menționat anterior, emulatorul implementat pentru testarea plutonului cu vehicule modelate cu dublu integrator a fost adaptat pentru noul model mai complex. Astfel, în blocul care simulează modelul vehiculului a fost apelat modelul (II.163). O altă modificare adusă emulatorului din Figura II.71 a fost introducerea unui bloc de întârziere (*Transport delay*), pe a treia mărime de intrare (i.e., traiectoria optimală primită de la predecesor) pentru a implementa perturbațiile pe canalul de comunicare.

II.3.3 Proiectarea unui sistem CACC cu reglarea după stare

Controlul cooperativ adaptiv al vitezei de croazieră (CACC) reprezintă o versiune avansată a sistemului ACC prin intermediul căreia vehiculele din pluton au posibilitatea de a transmite informații de la unul la altul utilizând o rețea wireless, sub denumirea de rețea ad-hoc pentru vehicule (VANET). Vehiculele primesc informații despre viteza vehiculelor din fața acestora. Acest sistem este încadrat la nivelul 2 de automatizare definit de SAE. Această îmbunătățire a sistemului ACC încearcă să crească eficiența acestuia fiindcă vehiculele vor accelera sau decelera cu anticipație față de vehiculul precedent pentru a putea fi obținute erori de distanță mai mici și pentru a evita eventualele coliziuni, considerând de asemenea o distanță de siguranță.

Se urmărește ca ACC să fie în primul rând un sistem de confort și apoi un sistem de siguranță și de aceea trebuie să fie adoptate distanțe între vehicule relativ mari. Când această distanță scade la o valoare mică de doar câțiva metri se așteaptă ca densitatea traficului să crească. Considerând cazul vehiculelor de transport greu, va apărea o reducere semnificativă a forței de tracțiune, care conduce la scăderea consumului de combustibil. Pentru a face posibilă menținerea stabilității plutonului, sistemul ACC are nevoie de o extensie, cum sunt comunicațiile între vehicule. Funcționalitatea CACC nu este încă disponibilă comercial chiar dacă oferă o multitudine de avantaje. Există multe lucruri care trebuie să fie considerate într-o implementare reală a unui pluton de vehicule încât este greu să se obțină un sistem foarte eficient și sigur considerând dezavantajele și problemele de configurare a rețelei de comunicații wireless [251].

Principalele componente care alcătuiesc un sistem CACC pot fi clasificate în următoarele trei funcții: comunicații între vehicule și infrastructură, care trebuie să creeze o legătură pentru vehicule pentru a transmite informații în timp real între acestea; interfețe CACC pentru a oferi suport șoferilor să reacționeze adecvat la funcționarea CACC în funcție de anumite condiții de condus; strategiile de control în vehicule trebuie să se ocupe cu determinarea acțiunilor potrivite pentru a menține în siguranță operațiile sistemului CACC [185].

Obiectivele care sunt urmărite de controlul automat al vitezei sunt următoarele: îmbunătățirile siguranței și a dinamicii traficului, creșterea capacității drumului prin adoptarea unor distanțe mai mici între vehicule, reducerea poluării și economisirea energiei prin scăderea forței de opunere aerodinamice și creșterea confortului șoferului. Un sistem CACC care este optimizat pentru a păstra o distanță constantă între vehicule nu este de natură să maximizeze eficiența consumului de combustibil când sunt necesare schimbări frecvente de viteză. Vehiculele echipate cu sisteme de control autonom reacționează folosind doar informațiile care sunt obținute prin intermediul dispozitivelor de percepere a schimbărilor (radar sau cameră), în timp ce vehiculele care dețin sisteme de control cooperative adaugă informațiile primite de la sistemul de comunicații între vehicule (V2V) la datele colectate de la senzori. Când comunicațiile V2V există în pluton, informațiile despre lider sunt primite prima dată de al doilea vehicul și ulterior când al treilea vehicul detectează reacția lui, datele sunt transmise către acesta care elaborează acțiunile necesare care trebuie executate. Astfel, toate informațiile sunt transmise de-a lungul șirului până la ultimul vehicul din pluton [186].

Sistemele tradiționale de management al traficului sunt construite ca și infrastructuri

centralizate echipate cu camere video și senzori care colectează datele despre starea traficului și le utilizează pentru a elabora deciziile adecvate. Implementarea unui astfel de sistem este foarte costisitoare și capacitatea de procesare și transferul de informații ale acestuia sunt extrem de importante. Ratele de transfer trebuie să fie foarte mari cu întârzieri mici și puține erori care pot fi propagate în rețea. Menținerea dispozitivelor care alcătuiesc acest sistem de comunicații este realizată periodic. În domeniul comunicațiilor, rețelele wireless au evoluat foarte rapid, care fac posibilă contruirea unei arhitecturi noi descentralizate bazate pe comunicațiile între vehicule. În acest caz, infrastructura este exclusă, sistemul fiind compus doar din vehicule [46].

În literatura de specialitate există numeroase lucrări care studiază sistemele ACC și mai recent și pe cele CACC. De exemplu, în [62] este prezentată proiectarea unui sistem ACC centralizat bazat pe controlul predictiv al modelului (MPC). În acea lucrare, se demonstrează că această metodă asigură stabilitatea plutonului de vehicule. Regulatorul propus este ales pentru a garanta integrarea constrângerilor sistemului pentru a reduce efectele întârzierilor și a pierderilor de date din rețea. Algoritmul a fost simulat utilizând un simulator în rețea LENA LET/EPC. O altă propunere a unui sistem CACC este reprezentată de abordarea creată de autorii lucrării [25]. În această lucrare, este considerat un pluton eterogen de vehicule pentru formularea problemei de control, care constă în urmărirea distanței dintre vehicule, viteză și accelerație. Pentru acest scop este propus un algoritm longitudinal descentralizat de urmărire. Elementele feed-forward conțin informații despre liderul plutonului și despre cel mai apropiat vehicul precedent, iar componentele feedback conțin stările vehiculelor gazdă. În [252], este prezentat un sistem CACC care constă în două regulatoare. Primul dintre acestea este utilizat pentru a trata cazul manevrei de apropiere de vehiculul din față, iar cel de-al doilea pentru a controla urmărirea după ce vehiculul se alătură plutonului.

Acest subcapitol are ca și scop principal proiectarea unui sistem CACC pentru un pluton de vehicule care poate fi folosit în cadrul infrastructurii de trafic actuale. Este considerată doar comunicarea cu vehiculul precedent care este implementată ca un semnal feed-forward. Acest tip de structură este denumită ACC semi-autonom [253], având avantajul că atunci când comunicațiile nu funcționează, funcționalitatea ACC standard va fi disponibilă. Obiectivul principal de control pentru un vehicul din pluton este să urmărească vehiculul precedent la o distanță dorită care depinde de viteza vehiculului. Metoda de spațiere dependentă de viteză se referă la distanța dintre vehicule cu înaintare în funcție de timp. Se presupune că liderul plutonului urmărește o referință de viteză variabilă în timp cu ajutorul sistemului CC. Celelalte vehicule ale plutonului utilizează dispozitive radar pentru a măsura distanțele dintre acestea și vehiculele din fața lor, care sunt folosite în regulatoare ACC standard de tip feedback. Viteza vehiculului precedent este disponibilă prin sistemul de comunicații wireless și este utilizată de un regulator feed-forward. În cadrul simulărilor este considerată și întârzierea recepției semnalului transmis.

II.3.3.1 Modelul de mișcare longitudinală a vehiculului

Forța de tracțiune dorită pentru deplasarea vehiculului poate fi obținută utilizând un tabel de căutare și prin aplicarea unghiului de accelerație potrivit sau a raportului potrivit

de injecție de combustibil. Se presupune că forța de frecare a anvelopelor nu reprezintă un factor de limitare. Astfel, se face presupunerea că forța de tracțiune poate fi manipulată în mod direct.

Considerând afirmațiile anterioare, ecuația de mișcare longitudinală a vehiculului are forma [138]:

$$m \frac{dv}{dt} = F_x - mg \sin \theta - fmg \cos \theta - 0.5\rho AC_d(v + v_w)^2, \quad (\text{II.195})$$

unde m este masa vehiculului, F_x este forța de tracțiune, g este accelerația gravitațională, θ este înclinarea drumului, f este coeficientul de rezistență la rulare, ρ este densitatea aerului, A este aria frontală a vehiculului, C_d este coeficientul de rezistență, v este viteza vehiculului și v_w este viteza vântului.

Ecuația (II.195) poate fi liniarizată prin considerarea unei condiții nominale de operare, când $dv/dt = 0$. În punctul de echilibru, ecuația (II.195) poate fi rezolvată pentru [138]:

$$F_{x0} = mg \sin \theta_0 + fmg \cos \theta_0 + 0.5\rho AC_d(v_0 + v_w)^2. \quad (\text{II.196})$$

Liniarizând ecuația (II.195) în funcție de starea de operare specificată folosind serii Taylor se obține următoarea relație:

$$\tau \dot{v}' + v' = K(u + w), \quad (\text{II.197})$$

unde variabilele incrementale sau perturbate împreună cu parametrii modelului sunt definite astfel:

$$\begin{aligned} v &= v_0 + v'; F_x = F_{x0} + u; \theta = \theta_0 + \theta', \\ w &= mg (f \sin \theta_0 - \cos \theta_0) \theta', \\ \tau &= (m / (\rho C_d A (v_0 + v_w))), \\ K &= (1 / (\rho C_d A (v_0 + v_w))). \end{aligned} \quad (\text{II.198})$$

Modelul vehiculului reprezentat în Figura II.74 a fost construit pe baza ecuațiilor liniarizate și poate fi exprimat ca o funcție de transfer:

$$G_v(s) = \frac{K}{\tau s + 1}. \quad (\text{II.199})$$

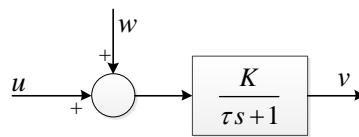


Figura II.74: Diagrama bloc a dinamicii longitudinale liniarizate a vehiculului

II.3.3.2 Formularea problemei de control a CACC

Un pluton de vehicule este considerat așa cum este ilustrat în Figura II.75. Obiectivul principal de control este urmărirea vehiculului precedent la o distanță dorită:

$$d_i^{ref} = r_i + h_{d,i} v_i, \quad (\text{II.200})$$

unde r_i este distanța dorită la staționare, $h_{d,i}$ este distanța dorită în funcție de viteza de deplasare, și v_i este viteza vehiculului i . Timpul de înaintare este timpul necesar vehiculului i pentru a ajunge la poziția curentă a vehiculului precedent $i-1$ când se deplasează cu o viteză constantă.

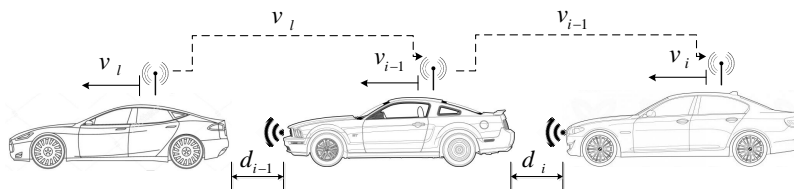


Figura II.75: Un pluton de vehicule cu sistem CACC

Mărimile valabile pentru vehiculul i dintr-un pluton sunt distanța d_i dintre vehiculele $i-1$ și i obținute cu un dispozitiv radar și viteza v_{i-1} a vehiculului precedent accesibilă prin comunicații wireless. Considerând obiectivul principal de control și mărimile cunoscute, o structură de control cu CACC a fost dezvoltată și este reprezentată în Figura II.76 pentru cele trei vehicule ilustrate în Figura II.75.

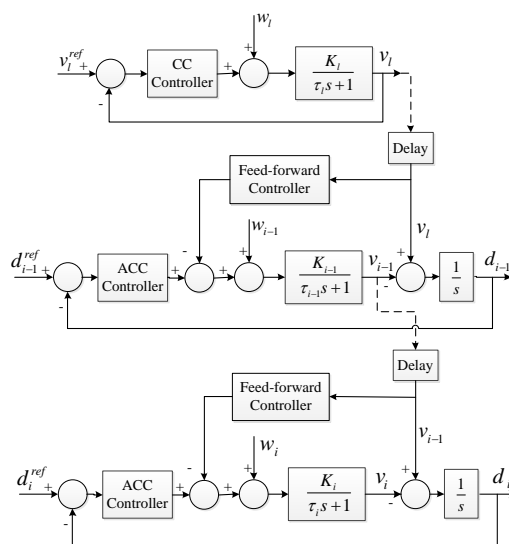


Figura II.76: Structura CACC pentru deplasarea în pluton

Se presupune că liderul plutonului va urmări o referință de viteză prestabilită variabilă în timp v_i^{ref} cu ajutorul unui sistem cruise control [254]. Fiecare vehicul urmăritor notat cu i în Figura II.75 și Figura II.76 are ca și mărimi specifice datele de ieșire ale radarului, care reprezintă distanța d_i față de vehiculul precedent, și viteza v_{i-1} a vehiculului din față obținută printr-un sistem de comunicații wireless. Având în vedere obiectivul principal de control de a urmări vehiculul precedent la o distanță dorită d_i^{ref} și considerând viteza v_{i-1} ca o perturbație măsurabilă, pentru fiecare vehicul urmăritor a fost adoptată o structură de control combinată între un regulator ACC și un regulator feed-forward.

II.3.3.3 Proiectarea sistemului CC pentru vehiculul lider

Dificultățile proiectării sistemului CC apar din cauza incertitudinilor introduse de variația masei vehiculului, și de perturbațiile externe apărute din cauza condiției drumului. Pentru sistemul CC a fost proiectat un regulator PI bazat pe modelul vehiculului (II.199) și utilizând o metodă de alocare a polilor. Considerând un regulator PI cu forma paralelă:

$$G_{cc}(s) = K_P + K_I/s \quad (\text{II.201})$$

și modelul vehiculului (II.199), funcția de transfer în buclă închisă este derivată după cum urmează:

$$G_{0cc}(s) = \frac{\frac{K_I K_I}{\tau_l} \left(\frac{K_P}{K_I} s + 1 \right)}{s^2 + \frac{K_I K_P + 1}{\tau_l} s + \frac{K_I K_I}{\tau_l}}. \quad (\text{II.202})$$

Regulatorul PI introduce un zero la funcția de transfer în buclă închisă care conduce la creșterea suprareglării sistemului de control. Zeroul introdus de regulatorul PI poate fi compensat prin introducerea unui bloc de anulare a zeroului pe calea feed-forward ca în Figura II.77, care deține următoarea funcție de transfer:

$$G_{zc}(s) = \frac{1}{\frac{K_P}{K_I} s + 1}. \quad (\text{II.203})$$

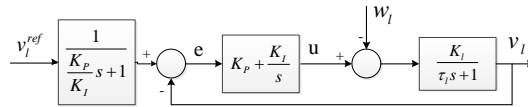


Figura II.77: Sistemul în buclă închisă cu regulator PI și bloc de anulare a zeroului pe calea feed-forward

Funcția de transfer în buclă închisă cu blocul de anulare a zeroului este echivalentă cu un sistem standard de ordinul doi. Utilizând această echivalență, parametrii de proporționalitate și de integrare ai regulatorului PI poate fi calculată astfel:

$$\begin{aligned} K_P &= (2\zeta\omega_n\tau_l - 1) / K_I, \\ K_I &= \tau_l\omega_n^2 / K_I, \end{aligned} \quad (\text{II.204})$$

unde ζ și ω_n sunt factorul de amortizare și frecvența naturală a sistemului de ordin doi. De obicei, parametrii sistemului de ordin doi sunt determinați din performanțele impuse cum sunt suprareglarea și timpul de răspuns.

II.3.3.4 Proiectarea sistemului de control pentru vehiculele urmăriitoare

Sistemul de control al fiecărui vehicul urmăritor deține două regulatoare: unul de tip feedback pentru menținerea distanțelor dorite dintre vehicule și altul feed-forward pentru

a compensa efectul perturbației măsurabile. Distanța dorită este menținută de un regulator ACC. Luând în considerare posibilitatea apariției unei erori la nivelul sistemului de comunicații wireless, regulatorul ACC trebuie să conțină o integrală dublă a erorii deoarece se presupune că viteza vehiculului precedent care acționează ca o perturbație poate fi modelată ca o intrare de tip rampă. Ca și rezultat, vehiculul controlat poate urmări vehiculul precedent chiar și atunci când acesta accelerează sau frânează brusc. Necesitatea existenței unui dublu integrator a făcut imposibilă folosirea unui regulator PID. Considerând aceasta, a fost folosit un algoritm de control bazat pe stare cu dublu integrator pentru regulatorul ACC.

Modelul de stare a părții fixate a vehiculului i pentru regulatorul ACC este obținut utilizând funcția de transfer (II.199) și prin adăugarea unui dublu integrator, rezultând astfel:

$$\begin{aligned}\dot{x}_1 &= -x_2 + v_{i-1}, \\ \dot{x}_2 &= -(1/\tau_i)x_2 + (K_i/\tau_i)(u + w_i), \\ \dot{x}_3 &= d_i^{ref} - d_i, \\ \dot{x}_4 &= x_3,\end{aligned}\tag{II.205}$$

unde $x_1 = d_i$, $x_2 = v_i$, x_3 este integrala erorii de control și x_4 este integrala dublă a erorii. Regulatorul bazat pe stare rezultat are forma:

$$u = -F_1x_1 - F_2x_2 - F_3x_3 - F_4x_4 = f^T x,\tag{II.206}$$

unde $f^T = -[F_1 \ F_2 \ F_3 \ F_4]$ și $x = [x_1 \ x_2 \ x_3 \ x_4]^T$.

Prin adăugarea regulatorului feed-forward G_{ff} pentru eliminarea perturbației măsurabile, rezultă structura de control din Figura II.78 pentru un vehicul următor.

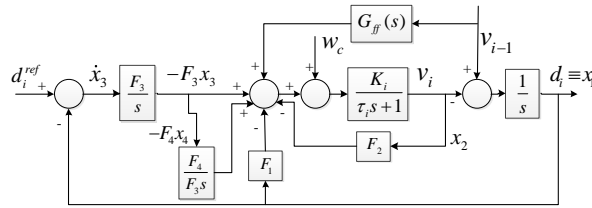


Figura II.78: Structura de control pentru un vehicul următor

Regulatorul feed-forward este proiectat pentru a reduce efectul perturbațiilor măsurabile rezultând [255]:

$$G_{ff}(s) = G_p^{-1}(s).\tag{II.207}$$

Regulatorul ideal feed-forward este determinat prin considerarea inversei dinamicii vehiculului (II.199), dar această inversă nu este realizabilă și de aceea va fi folosită o aproximație. Astfel, inversa funcției de transfer (II.199) poate fi aproximată cu:

$$G_p^{-1}(s) \cong \frac{1 + s\tau_i}{K_i(1 + s\tau_i/N)},\tag{II.208}$$

unde N furnizează intervalul frecvenței unde inversiunea este validă.

Matricea de stare a regulatorului ACC (II.206) poate fi calculată utilizând formula Ackermann. Modelul de stare (II.205) a vehiculului i are forma:

$$\begin{aligned} \dot{x} &= Ax + bu + Dw^T, \\ y &= c^T x, \end{aligned} \quad (\text{II.209})$$

cu matricele/vectorii:

$$\begin{aligned} A &= \begin{bmatrix} 0 & -1 & 0 & 0 \\ 0 & -1/\tau_i & 0 & 0 \\ -1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}, b = \begin{bmatrix} 0 \\ K_i/\tau_i \\ 0 \\ 0 \end{bmatrix}, D = \begin{bmatrix} 1 & 0 \\ 0 & K_i/\tau_i \\ 0 & 0 \\ 0 & 0 \end{bmatrix}, \\ c^T &= [1 \ 0 \ 0 \ 0], w^T = [v_{i-1} \ w_i]. \end{aligned} \quad (\text{II.210})$$

Utilizând formula Ackermann, se obține următoarea matrice de stare:

$$f^T = [0 \ 0 \ 0 \ 1] R^{-1} P_c(A), \quad (\text{II.211})$$

unde $R = [b \ Ab \ A^2b \ A^3b]$ este matricea de controlabilitate și

$P_c(s) = (s^2 + 2\zeta\omega_n s + \omega_n^2)(s + 4\omega_n)^2$ este polinomul caracteristic al sistemului în buclă închisă construit pe baza performanțelor dorite. De exemplu, prin impunerea suprareglării și a timpului de răspuns, pot fi obținute factorul de amortizare ζ și frecvența naturală ω_n .

Structura CACC din Figura II.78 utilizează comunicații între vehicule pentru a obține informații despre viteza vehiculului precedent considerată ca și perturbație măsurabilă care este eliminată de regulatorul feed-forward. De obicei, informațiile între vehicule sunt transmise o dată la fiecare 100 ms. Din acest motiv, apare o întârziere în sistemul de control, a cărei valoare maximă va fi de 100 ms.

II.3.3.5 Simulator Matlab pentru un pluton cu CACC

Abordarea propusă pentru proiectarea sistemului CACC utilizat pentru deplasarea în pluton a vehiculelor a fost evaluat prin simulare utilizând MATLAB/Simulink [254]. În Figura II.79, este ilustrat un pluton de cinci vehicule cu un lider și patru urmăritori. Pentru simulare au fost folosite următoarele valori ale parametrilor din ecuația (II.195):

$$\begin{aligned} g &= 9.81m/s, v_0 = 25m/s, \theta_0 = 5, m = 1000kg, \\ \rho &= 1.202kg/m^3, A = 1.5m^2, C_d = 0.5, \\ f &= 0.015, v_w = 2m/s. \end{aligned} \quad (\text{II.212})$$

Structura de control a vehiculului lider (Figura II.80) este implementată cu sistemul de control a vitezei de croazieră prezentat în Figura II.77. Proiectarea regulatorului PI este realizată cu metoda alocării polilor, rezultând parametrii de proiectare $K_P = 9695.7$, $K_I = 29160$. Figura II.81 ilustrează sistemul de control a vehiculelor urmăritoare construite cu structurile de control feedback-feedforward din Figura II.76. Algoritm ACC constă într-un regulator cu reglare după stare cu patru stări, două dintre acestea fiind introduse cu blocuri integratoare aplicate erorii de distanță (Figura II.78). Regulatorul este proiectat utilizând tehnica de stabilire a locului polilor, rezultând $f^T = [-3010000 \ 90000 \ 38680000$

184390000]. Regulatorul feed-forward a fost proiectat utilizând relațiile (II.207) și (II.208). Valorile parametrilor de proiectare pentru regulatoarele PI și ACC au valori mari fiindcă elementele de execuție lipsesc din structura de control. Elementele de execuție ar trebui să aibă rolul de a amplifica semnalele furnizate de regulatoare și să genereze forța de tracțiune. În diagrama simplificată utilizată în acest raport forța de tracțiune este transmisă direct de regulatoare.

În diagrama bloc din Figura II.79 există două posibilități de a stabili referința de distanță. Prima este reprezentată de o valoare constantă (distance-headway) $d_i^{ref} = 4m$ introdusă cu un bloc Step, iar cea de-a doua (time-headway) este calculată conform ecuației (II.200) depinzând de viteza fiecărui vehicul și de o valoare predefinită de distanță $r_i = 1m$. Parametrul $h_{d,i}$ este setat la 0.1s. Datele sunt eșantionate cu o rată de $T_s = 0.01s$.

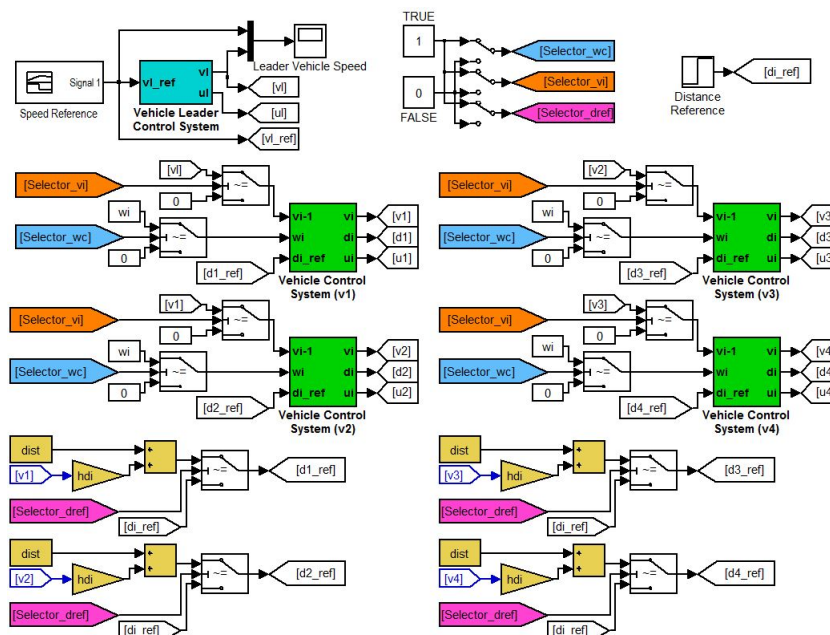


Figura II.79: Reprezentarea Simulink a plutonului de vehicule

Vehiculele urmăritoare trebuie să răspundă la acțiunile predecesorilor săi mai rapid decât vehiculul lider răspunde la adaptarea automată a propriei viteze. Astfel timpul de răspuns setat pentru celelalte vehicule este semnificativ mai mic decât cel stabilit pentru liderul plutonului.

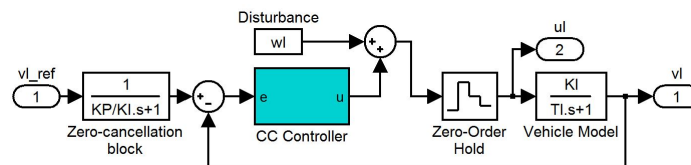


Figura II.80: Structura Simulink a sistemului de control a vitezei vehiculului lider

Toate vehiculele din plutonul considerat sunt identice. Semnalele de viteză ale fiecărui vehicul din pluton sunt ilustrate în Figura II.82. Se poate observa că fiecare sistem de

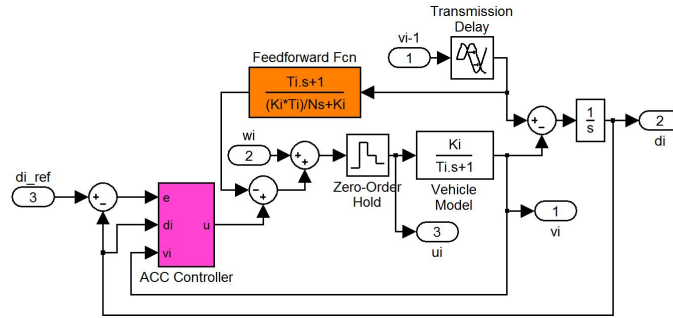


Figura II.81: Schema Simulink a sistemului de control pentru vehiculele urmăritoare

control acționează în așa fel încât vehiculele sunt capabile să-și mențină propriile viteze în funcție de vitezele predecesorilor acestora. În Figura II.83, sunt reprezentate distanțele dintre vehicule după ce a fost aplicată o referință dependentă de viteză asupra intrărilor sistemelor de control. Se observă că în acest caz formele de undă ale distanțelor urmăresc profilul de viteză introdus ca referință pentru lider. Distanța în staționare este considerată egală cu un metru. Când viteza vehiculului crește, distanța dintre un vehicul și vehiculul precedent crește de asemenea. Această caracteristică proporțională este aplicată și atunci când viteza scade.

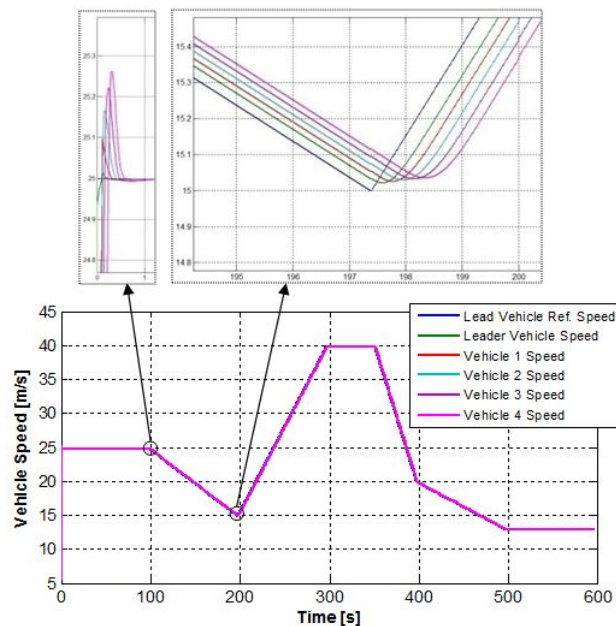


Figura II.82: Vitezele tuturor vehiculelor din pluton

Figura II.84 ilustrează distanțele între vehicule obținute când referința este setată la o valoare constantă. Distanța dorită în acest caz este egală cu 4 metri și se arată că o distanță constantă este menținută de fiecare vehicul independent de viteza acestuia.

Prin intermediul răspunsurilor din Figura II.83 s-a dovedit că fiecare vehicul din plutonul proiectat este capabil să adapteze corespunzător viteza proprie încât să mențină o distanță

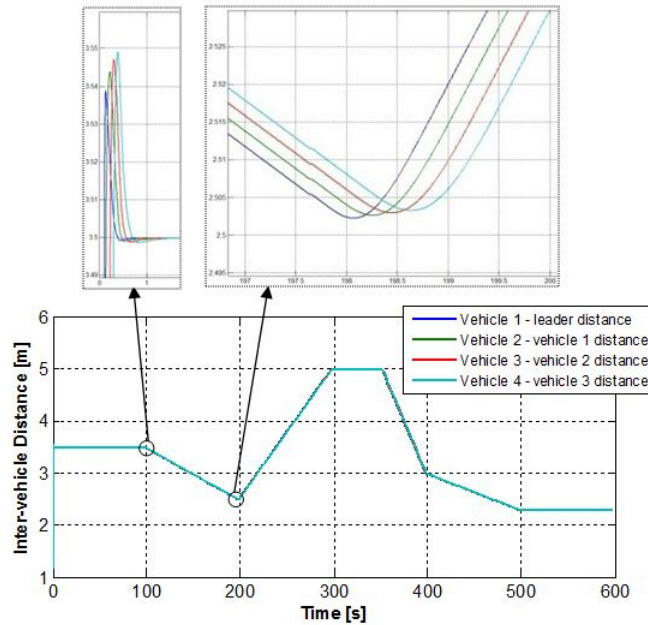


Figura II.83: Distanțele dintre vehicule cu referință dependentă de viteză

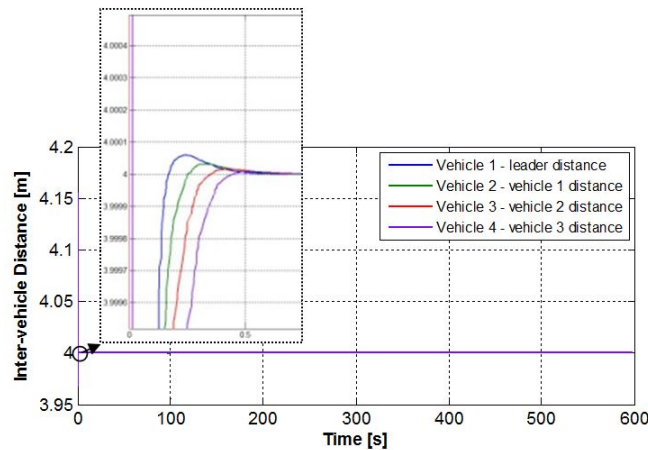


Figura II.84: Distanțele dintre vehicule cu o referință constantă

de siguranță între acesta și predecesorul lui. Acest lucru este necesar în situațiile de trafic real pe autostrăzi pentru a evita coliziunile cu vehiculul din față deoarece, cu cât viteza este mai mare, cu atât distanța de frânare crește. Cazul distanței constante, ilustrat în Figura II.84, este prezentat în scop teoretic pentru a dovedi că vehiculele pot menține de asemenea o distanță constantă. Efectele perturbațiilor introduse de rețeaua de comunicații wireless și de dinamica vehiculului sunt compensate. Alți factori care pot afecta comportamentul sistemelor CACC sunt viteza vântului (v_w) și înclinarea drumului (θ). Pentru plutonul proiectat, răspunsurile obținute sunt aceleași ca în Figura II.82 și Figura II.83 după ce câteva valori aleatorii au fost considerate pentru factorii măsurabili menționați anterior. În Figura II.85, forțele de tracțiune calculate de regulatorul CC și de sistemele CACC sunt prezentate.

Valorile forței de tracțiune sunt proporționale cu vitezele vehiculelor ilustrate în Figura II.82 în funcție de cazul accelerării sau decelerării forța crește sau scade. Când este necesară menținerea unei valori de viteză constantă forța de tracțiune este constantă.

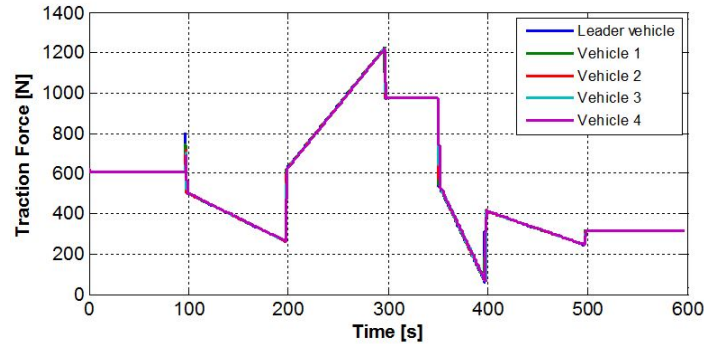


Figura II.85: Forțele de tracțiune ale fiecărui vehicul din pluton

II.3.4 Proiectarea unui pluton cu controlul predictiv bazat pe modelul în spațiul stărilor

Sistemul liniar continuu în timp în spațiul stărilor este de forma:

$$\begin{aligned} \dot{x} &= A_c x + B_c u \\ y &= C_c x + D_c u \\ z &= H_c x \end{aligned} \quad (\text{II.213})$$

În cele mai multe cazuri, $H=C$, și astfel rezultă că $y=z$.

În urma discretizării ecuațiilor (II.213) se obține:

$$\begin{aligned} x(k+1) &= Ax(k) + Bu(k) \\ y(k) &= Cx(k) \\ z(k) &= Hx(k) \end{aligned} \quad (\text{II.214})$$

Se fac următoarele presupuneri:

- perechea (A,B) este stabilizabilă și perechea (C,A) este detectabilă
- $C=I$ conduce la reglarea după stare
- $H=C$ indică faptul că ieșirile și stările sunt variabile controlabile

Cunoscând modelul discret (II.214) problema existentă este proiectarea legii de control cu reglarea după stare $u(k) = Kx(k)$ astfel încât originea sistemului în buclă închisă

$$x(k+1) = (A + BK)x(k) \quad (\text{II.215})$$

este asimptotic stabilă, ceea ce necesită ca $(A+BK)$ să fie stabilă.

II.3.4.1 Controlul predictiv fără restricții

Se urmărește determinarea unei secvențe de intrare care să minimizeze funcția de cost cu orizont finit [256]:

$$V(x, (u_0, \dots, u_{N-1})) = x_N^T P x_N + \sum_{i=0}^{N-1} (x_i^T Q x_i + u_i^T R u_i) \quad (\text{II.216})$$

unde

$$\begin{aligned} x_0 &= x \\ x_{i+1} &= A x_i + B u_i, i = 0, 1, \dots, N-1 \end{aligned} \quad (\text{II.217})$$

Se definesc vectorii de stivă:

$$U = \begin{pmatrix} u_0 \\ u_1 \\ u_2 \\ \vdots \\ u_{N-1} \end{pmatrix}, X = \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ \vdots \\ x_N \end{pmatrix} \quad (\text{II.218})$$

în care u_i reprezintă secvența de intrare, iar x_i secvența de stare.

Funcția de cost este definită astfel:

$$V(x, U) = x_N^T P x_N + \sum_{i=0}^{N-1} (x_i^T Q x_i + u_i^T R u_i) \quad (\text{II.219})$$

Funcția de valoare are forma:

$$V^*(x) = \min_U V(x, U) \quad (\text{II.220})$$

Secvența optimală de intrare se definește după cum urmează:

$$U^*(x) = \arg \min_U V(x, U) = \{u_0^*(x), u_1^*(x), \dots, u_{N-1}^*(x)\} \quad (\text{II.221})$$

Se calculează matricile de predicție Φ și Γ astfel încât:

$$X = \Phi x + \Gamma U \quad (\text{II.222})$$

Funcția de cost $V(\cdot)$ se rescrie în funcție de x și U . Apoi pentru aceasta se calculează gradientul $\nabla_U V(x, U)$. Se consideră $\nabla_U V(x, U) = 0$ și se rezolvă pentru $U^*(x)$.

Legea de control MPC este reprezentată de prima parte a secvenței optime de intrare:

$$u_0^*(x) = \begin{pmatrix} I_m & 0 & \cdots & 0 \end{pmatrix} U^*(x) \quad (\text{II.223})$$

Se urmărește determinarea matricilor Φ și Γ pentru a se îndeplini ecuația (II.222). Se scrie forma extinsă pentru cele N stări ale sistemului:

$$\begin{aligned} x_1 &= A x_0 + B u_0 \\ x_2 &= A(A x_0 + B u_0) + B u_1 = A^2 x_0 + A B u_0 + B u_1 \\ &\vdots \\ x_N &= A^N x_0 + A^{N-1} B u_0 + \cdots + A B u_{N-2} + B u_{N-1} \end{aligned} \quad (\text{II.224})$$

Se determină forma matriceală a sistemului după cum urmează:

$$\begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{pmatrix} = \begin{pmatrix} A \\ A^2 \\ \vdots \\ A^N \end{pmatrix} x_0 + \begin{pmatrix} B & 0 & \cdots & 0 \\ AB & B & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{N-1}B & A^{N-2}B & \cdots & B \end{pmatrix} \begin{pmatrix} u_0 \\ u_1 \\ \vdots \\ u_{N-1} \end{pmatrix} \quad (\text{II.225})$$

Din relația (II.225) pot fi determinate matricile Φ și Γ :

$$\Phi = \begin{pmatrix} A \\ A^2 \\ \vdots \\ A^N \end{pmatrix}, \Gamma = \begin{pmatrix} B & 0 & \cdots & 0 \\ AB & B & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{N-1}B & A^{N-2}B & \cdots & B \end{pmatrix} \quad (\text{II.226})$$

Funcția de cost poate fi rescrisă astfel:

$$\begin{aligned} V(x, U) &= x_N^T P x_N + \sum_{i=0}^{N-1} (x_i^T Q x_i + u_i^T R u_i) \\ &= x_0^T Q x_0 + \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{pmatrix}^T \begin{pmatrix} Q & & & \\ & Q & & \\ & & Q & \\ & & & P \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_N \end{pmatrix} \\ &\quad + \begin{pmatrix} u_0 \\ u_1 \\ \vdots \\ u_{N-1} \end{pmatrix}^T \begin{pmatrix} R & & & \\ & R & & \\ & & R & \\ & & & R \end{pmatrix} \begin{pmatrix} u_0 \\ u_1 \\ \vdots \\ u_{N-1} \end{pmatrix} \\ &= x^T Q x + X^T \Omega X + U^T \Psi U \end{aligned} \quad (\text{II.227})$$

Luând în considerare ecuațiile următoare:

$$\begin{aligned} V(x, U) &= x^T Q x + X^T \Omega X + U^T \Psi U \\ X &= \Phi x + \Gamma U \end{aligned} \quad (\text{II.228})$$

noua formă a funcției de cost este:

$$\begin{aligned} V(x, U) &= x^T (Q + \Phi^T \Omega \Phi) x + U^T (\Psi + \Gamma^T \Omega \Gamma) U + 2U^T \Gamma^T \Omega \Phi x \\ &= \frac{1}{2} U^T G U + U^T F x + x^T (Q + \Phi^T \Omega \Phi) x \end{aligned} \quad (\text{II.229})$$

unde

$$\begin{aligned} G &= 2(\Psi + \Gamma^T \Omega \Gamma) \succ 0 \\ F &= 2\Gamma^T \Omega \Phi \end{aligned} \quad (\text{II.230})$$

Secvența optimă de intrare este:

$$U^*(x) = -G^{-1} F x \quad (\text{II.231})$$

Considerând legea de control MPC (II.223) se definește

$$K_{MPC} = - (I_m \ 0 \ \cdots \ 0) G^{-1} F \quad (\text{II.232})$$

astfel încât $u = K_{MPC} x$ pentru cazul controlului predictiv fără restricții.

II.3.4.2 Controlul predictiv cu restricții

La fel ca în cazul controlului fără restricții și pentru controlul predictiv cu restricții se dorește minimizarea funcției de cost (II.216) care garantează satisfacerea tuturor constrângerilor asupra orizontului de predicție $i=0,1,\dots,N$ [256].

Se consideră forma matriceală a sistemului (II.225) în care se regăsesc stările predictate X sub forma din relația (II.222).

Un set liniar de constrângeri de inegalitate ale stărilor predictate x_i și ale intrărilor u_i se încorporează în sistem:

$$\begin{aligned} M_i x_i + E_i u_i &\leq b_i \\ M_N x_N &\leq b_N \end{aligned} \quad (\text{II.233})$$

Pentru simplitate se presupune că $E_i = E$, $M_i = M$ și $b_i = b$ pentru $i=0,1,\dots,N-1$.

Se presupune că există următoarele constrângeri pentru intrare și ieșire:

$$\begin{aligned} u_{low} &\leq u_i \leq u_{high}, i = 0, 1, \dots, N - 1 \\ y_{low} &\leq y_i \leq y_{high}, i = 0, 1, \dots, N \end{aligned} \quad (\text{II.234})$$

Considerând că $y_i = Cx_i$ constrângerile (II.234) se pot rescrie astfel:

$$\begin{pmatrix} 0 \\ 0 \\ -C \\ C \end{pmatrix} x_i + \begin{pmatrix} -I \\ I \\ 0 \\ 0 \end{pmatrix} u_i \leq \begin{pmatrix} -u_{low} \\ u_{high} \\ -y_{low} \\ y_{high} \end{pmatrix}, i = 0, 1, \dots, N - 1 \quad (\text{II.235})$$

Făcându-se o identificare între relațiile (II.233) și (II.235) se obțin matricile:

$$\begin{aligned} M_i &= \begin{pmatrix} 0 \\ 0 \\ -C \\ C \end{pmatrix}, E_i = \begin{pmatrix} -I \\ I \\ 0 \\ 0 \end{pmatrix}, b_i = \begin{pmatrix} -u_{low} \\ u_{high} \\ -y_{low} \\ y_{high} \end{pmatrix}, i = 0, 1, \dots, N - 1 \\ M_N &= \begin{pmatrix} -C \\ C \end{pmatrix}, b_N = \begin{pmatrix} -y_{low} \\ y_{high} \end{pmatrix} \end{aligned} \quad (\text{II.236})$$

Luând în considerare toate constrângerile se obține forma matriceală următoare:

$$\begin{aligned} &\begin{pmatrix} M_0 \\ 0 \\ \vdots \\ 0 \end{pmatrix} x_0 + \begin{pmatrix} 0 & \cdots & 0 \\ M_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & M_N \end{pmatrix} \begin{pmatrix} x_1 \\ \vdots \\ x_N \end{pmatrix} \\ &+ \begin{pmatrix} E_0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & E_{N-1} \\ 0 & \cdots & 0 \end{pmatrix} \begin{pmatrix} u_0 \\ \vdots \\ u_{N-1} \end{pmatrix} \leq \begin{pmatrix} b_0 \\ b_1 \\ \vdots \\ b_N \end{pmatrix} \end{aligned} \quad (\text{II.237})$$

Prin aproximare se pot defini Δ , Υ , ε și c și astfel rezultă:

$$\Delta x + \Upsilon X + \varepsilon U \leq c \quad (\text{II.238})$$

Înlocuind $X = \Phi x + \Gamma U$ în (II.238) constrângerile pot fi rescrise astfel:

$$JU \leq c + Wx \quad (\text{II.239})$$

unde

$$\begin{aligned} J &= \Upsilon \Gamma + \varepsilon \\ W &= -\Delta - \Upsilon \Phi \end{aligned} \quad (\text{II.240})$$

Având matricile Q și A și vectorii c și b , problema de optimizare:

$$\min_{\theta} \frac{1}{2} \theta^T Q \theta + c^T \theta \quad (\text{II.241})$$

cu $A\theta \leq b$ este numită problemă pătratică.

În Matlab problema pătratică poate fi rezolvată utilizând:

$$U = \text{quadprog}(G, Fx, J, c + Wx) \quad (\text{II.242})$$

Legea de control MPC este reprezentată de prima parte a secvenței optime de intrare:

$$K_{MPC} := u_0^*(x) = \begin{pmatrix} I_m & 0 & \cdots & 0 \end{pmatrix} U^*(x) \quad (\text{II.243})$$

Având în vedere că $U^*(x)$ nu mai este liniar, legea de control este neliniară.

II.3.4.3 Simulator Matlab pentru un pluton de vehicule cu MPC cu restricții

Modelul vehiculului utilizat pentru proiectarea plutonului de vehicule cu MPC este cel din relația (II.199) cu parametrii (II.198). De asemenea valorile folosite pentru toți parametrii specifici ai vehiculelor sunt cele din (II.212).

În Figura II.86 este ilustrată o structură Simulink al unui pluton cu 15 vehicule în care fiecare vehicul urmăritor are ca și mărime de intrare viteza vehiculului anterior. Pentru vehiculul lider se introduce la intrare un profil de viteză cu multiple accelerări și decelerări pentru a simula o situație reală de trafic. Figura II.87 ilustrează structura de control a vehiculului lider în care se utilizează un regulator PI pentru controlul vitezei construindu-se astfel un sistem Cruise Control clasic.

În Figura II.88 se poate observa schema de control a vehiculelor urmăritoare care cuprinde o structură feedback-feedforward în care viteza vehiculului precedent este introdusă ca o perturbare măsurabilă. Regulatorul MPC proiectat are ca și intrări poziția și viteza vehiculului controlat, precum și distanța dintre acesta și vehiculul anterior. Structura detaliată a regulatorului MPC este vizibilă în Figura II.89 ce conține și funcția pentru rezolvarea problemei pătratice (*quadprog()*) într-un bloc de tipul Matlab Embedded Function.

Formele de undă ale vitezelor vehiculelor din pluton sunt ilustrate în Figura II.90 și se poate observa că toate semnalele urmăresc forma vitezei de referință a vehiculului lider. Cel mai bine se observă în momentul în care anumite porțiuni din grafic sunt mărite. Comportamentul sistemului este cel dorit, în acest caz, deoarece nu apar suprareglări sau alte efecte nedorite pe parcursul simulării. Și în cazul distanțelor dintre vehicule rezultatele sunt unele dorite. În funcție de vitezele cu care vehiculele din pluton se deplasează distanțele dintre

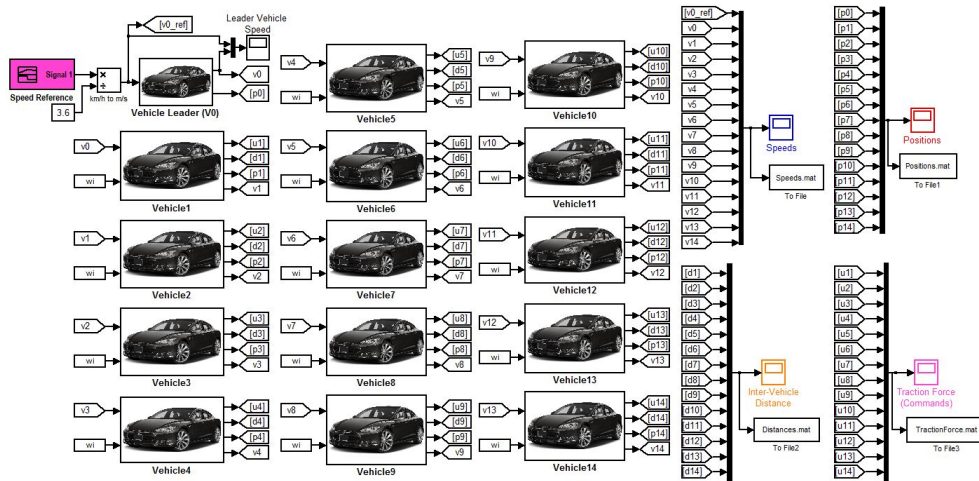


Figura II.86: Un pluton cu 15 vehicule în Simulink

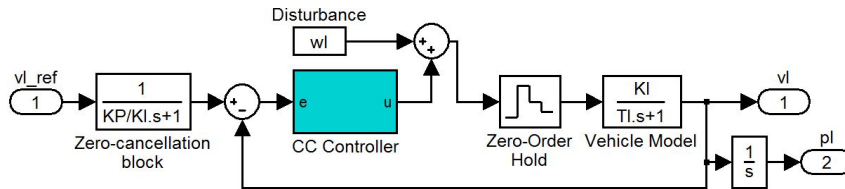


Figura II.87: Structura de control a vehiculului lider

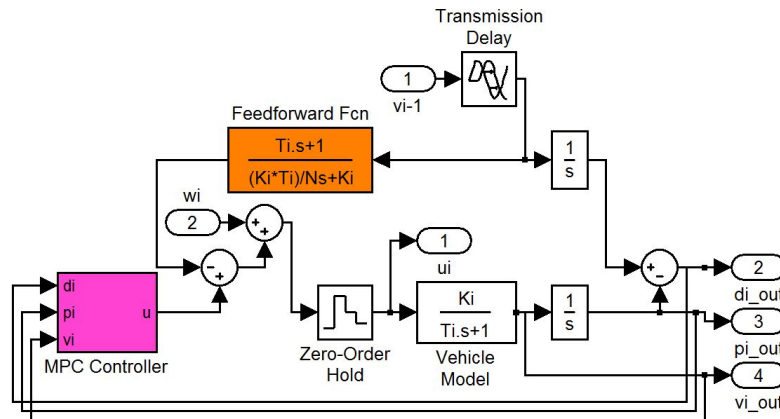


Figura II.88: Structura de control a vehiculelor urmăritoare

acestea variază. Pentru o viteză mai mare, distanța va fi mai mare pentru ca vehiculele să poată frâna în timp util, în caz de urgență, fără a se produce coliziuni între acestea. Figura II.91 ilustrează comportamentul plutonului în ceea ce privește distanțele dintre vehicule.

De asemenea pozițiile vehiculelor din pluton ilustrate în Figura II.92 demonstrează că acestea sunt aliniate la distanțe egale unele de altele și se deplasează într-un grup organizat.

Forțele de tracțiune generate de reglatoarele MPC ale vehiculelor urmăritoare, așa cum

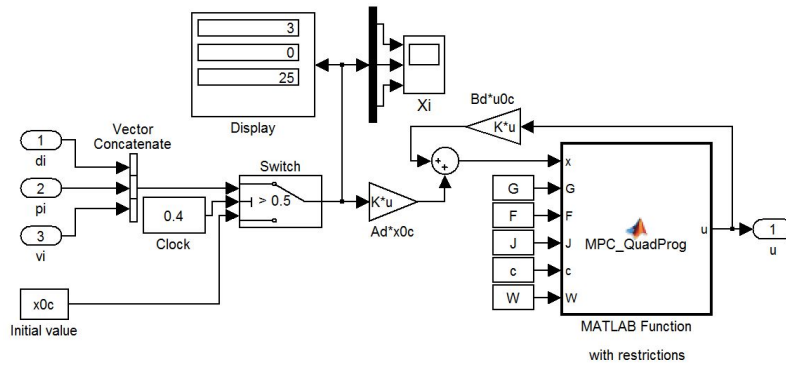


Figura II.89: Regulatorul MPC al fiecărui vehicul următor

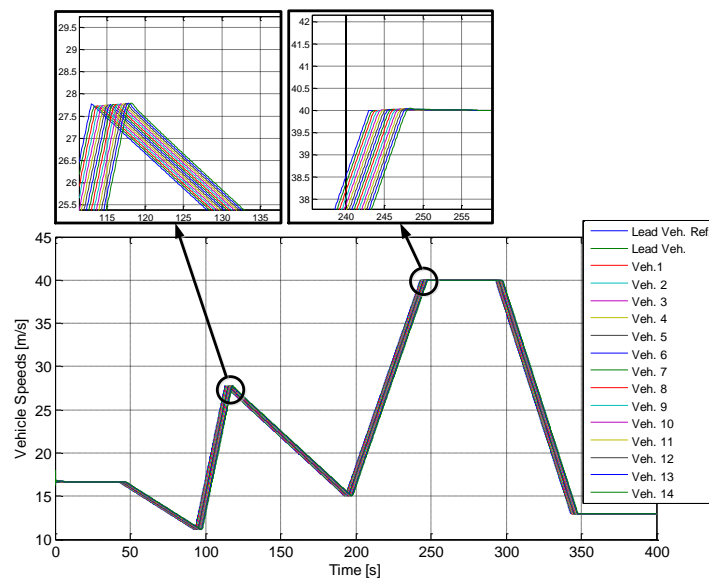


Figura II.90: Vitezele vehiculelor din pluton

sunt ilustrate în Figura II.93, reacționează la evoluțiile vitezei vehiculului lider după cum urmează: când liderul frânează forța de tracțiune (ieșirea de comandă a regulatorului) scade, iar când acesta accelerează forța crește.

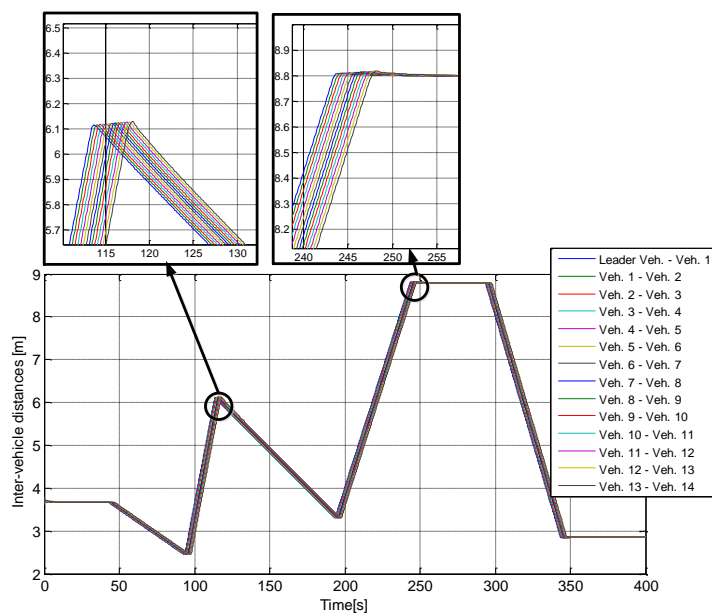


Figura II.91: Distanțele dintre vehiculele din pluton

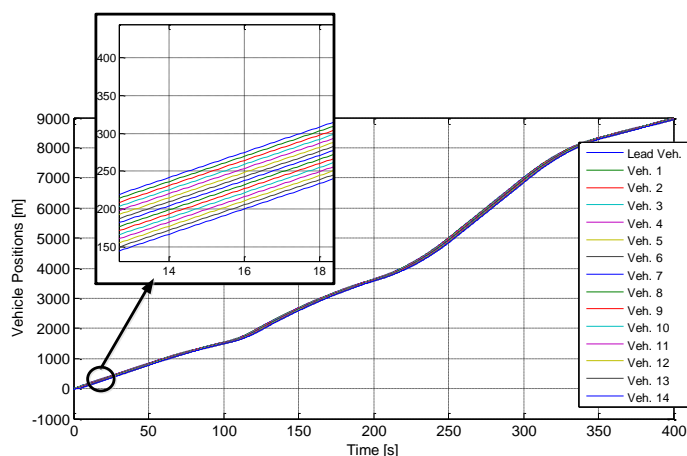


Figura II.92: Pozițiile vehiculelor din pluton

II.4 (A2.3) Verificarea stabilității plutonului de vehicule cooperante bazate pe multiagenți

În această secțiune, sunt definite condițiile de stabilitate pentru algoritmul DMPC prezentat în secțiunea II.2, pornind de la rezultatele propuse în [257], original formulate pentru o strategie de control predictiv centralizată definită pentru procese modelate prin funcții de transfer intrare-ieșire. Fie următoarele ipoteze aplicate pentru algoritmul DMPC:

- predicțiile sunt facute pentru procese modelate prin funcții de transfer intrare-ieșire augmentate cu modele pentru perturbații

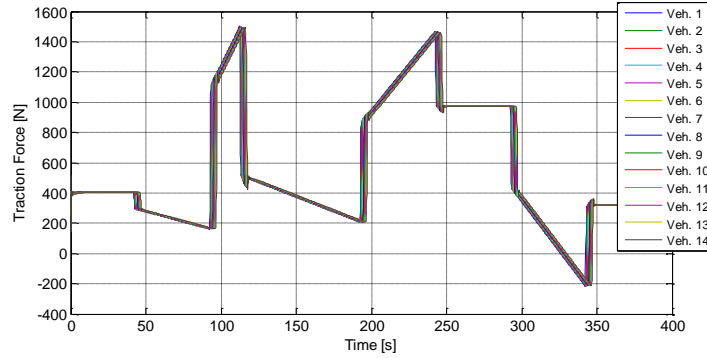


Figura II.93: Forțele de tracțiune ale vehiculelor

- orizontul de control și orizontul de predicție nu sunt egale
- nu sunt impuse restricții terminale
- nu se minimizează o funcție de cost terminală
- nu este definită o lege de control terminală

Pentru fiecare sub-sistem i , $i \in \mathcal{N}$, problema de control optimală, cu restricții și orizont finit, descrisă original în formulare intrare-ieșire, se rescrie într-o formulare cu model intrare-ieșire după cum urmează [257]:

$$\min_{u_i} \sum_{l=0}^{N_p} (r_i(k+l|k) - C_i x_i(k+l|k))^2 + \alpha_i \cdot \sum_0^{N_c-1} u_i(k+l|k)^2 \quad (\text{II.244})$$

s.t.

$$\Delta u_i(k+l|k) = 0, \forall l \in [N_c, N_p] \quad (\text{II.245})$$

$$\begin{aligned} \Delta u_i &= u_i(k+l|k) - u_i(k+l-1|k) \\ x_i^{\min} &\leq x_i(k+l|k) \leq x_i^{\max} \in \mathbb{X}, \forall l \in [1, N_p] \end{aligned} \quad (\text{II.246})$$

unde prin α_i se ponderează variabila de control.

Astfel, toate definițiile și restricțiile formulate pentru modelele intrare-iesire trebuie rescrise pentru modelele intrare-stare-iesire. Întrucât algoritmul DMPC definit cu model intrare-stare-iesire se conformează cu ultimele patru ipoteze, condițiile de stabilitate ce urmează să fie definite rămân valide. De asemenea, funcția cost (II.244) are aceeași structură cu funcția de cost DMPC (II.25), cu excepția faptului că în funcția de cost DMPC, variabila de control este definită sub formă incrementală. Pentru a evita complicarea notațiilor în urmă toarele definiții și teoreme, nu se vor folosi indici inferiori corespunzatori variabilelor fiecărui sub-sistem (i.e., x_i se va nota cu x).

Definiția II.4.1 Setul robust atins \tilde{R} este setul stărilor la care sistemul va evolua la următoarea perioadă de eșantionare, pornind de la starea inițială $x_i(k)$, mărimea de intrare admisibilă

și perturbațiile admise:

$$\begin{aligned}\tilde{R} &= x_i(k+1) \in \mathfrak{R}^n | \exists x_i(k) \in \mathbb{X}, u_i(k) \in U_i, n_i(k) \in \mathbb{W} \\ x_i(k+1) &= f(x_i(k), u_i(k), n_i(k))\end{aligned}\quad (\text{II.247})$$

unde indicele i corespunde sub-systemului.

Definiția II.4.2 Setul robust controlabil în v -pași \tilde{K}_v este setul stărilor în \mathbb{X} care poate fi condus de o secvență de intrări de lungime v că tre un set arbitrar țintă T în exact v pași, garantând evoluția stărilor în \mathbb{X} pentru $v-1$ pași pentru toate perturbațiile:

$$\begin{aligned}\tilde{K}_v &= x_i(k) \in \mathfrak{R}^n | \exists u_i(k+l|k) \in U_i|_0^{v-1} \\ x_i(k+l|k) &\in \mathbb{X}|_{l=1}^{v-1} \in \mathbb{X} \\ x_i(k+v|k) &\in T, \forall \{n_i(k+l|k) \in \mathbb{W}\}|_{l=0}^{v-1}\end{aligned}\quad (\text{II.248})$$

unde indicele i corespunde sub-systemului.

Definiția II.4.3 Setul robust admisibil în v -pași \tilde{C}_v este setul stărilor în \mathbb{X} pentru care există o secvență admisibilă de control de lungime v , care garantează evoluția stărilor în \mathbb{X} pentru v pași, pentru toate perturbațiile:

$$\begin{aligned}\tilde{C}_v &= x_i(k) \in \mathfrak{R}^n | \exists u_i(k+l|k) \in U_i|_0^{v-1} \\ x_i(k+l|k) &\in \mathbb{X}|_{l=1}^v \in \mathbb{X} \\ \forall \{n_i(k+l|k) &\in \mathbb{W}\}|_{l=0}^{v-1}\end{aligned}\quad (\text{II.249})$$

Definiția II.4.4 Un set robust tunel în v -pași $\tilde{L}(\mathbb{X})$ este un set robust admisibil în v -pași $\tilde{C}_v(\mathbb{X})$ cu restricția că secvența de control admisibilă rămâne constantă :

$$\tilde{L}_v = \tilde{C}_v(\mathbb{X}) | \{\Delta u_i(k+l) = 0\}_{l=1}^{v-1}\quad (\text{II.250})$$

Definiția II.4.5 Setul robust fezabil \tilde{X}_F este setul stărilor pentru care există o secvență de control admisibilă care satisface restricțiile pentru stări și ieșiri pentru toate secvențele de perturbații.

Teorema II.4.6 Setul robust fezabil $\tilde{X}_F(\mathbb{X}, N_c, N_p)$ este dat de

$$\tilde{X}_F(\mathbb{X}, N_c, N_p) = \tilde{K}_{N_c}(\mathbb{X}, \tilde{L}_{N_p-N_c}(\mathbb{X}))\quad (\text{II.251})$$

unde \tilde{L} este setul robust tunel, \mathbb{X} este setul de stări admisibile, \tilde{K} este setul robust controlabil.

Demonstrație II.4.7 Construirea setului robust fezabil poate fi împărțită temporal în două părți de la sfârșit că tre început. în orizontul $N_p - N_c$ se cere păstrarea secvenței de control constantă într-un set admisibil \tilde{K} care satisface setul de restricții \mathbb{X} , pentru toate perturbațiile. Aceasta reprezintă setul robust tunel dat de $\tilde{L}_{N_p-N_c}(\mathbb{X})$. Cerința pe durata orizontului de comandă N_c este să se găsească secvența de control care satisface restricțiile pentru stări și este în interiorul setului țintă $\tilde{L}_{N_p-N_c}(\mathbb{X})$. Acesta este dat de setul robust controlabil $\tilde{K}_{N_c}(\mathbb{X}, \tilde{L}_{N_p-N_c}(\mathbb{X}))$ care satisface condiția de a fi și un set robust fezabil.

Definiția II.4.8 Problema de optimizare (II.244) este robustă persistentă fezabilă *dacă și numai dacă* starea inițială și evoluțiile viitoare aparțin setului robust fezabil.

Propoziția II.4.9 *Dacă setul robust fezabil \tilde{X}_F este mărginit și problema de optimizare (II.244) este robust persistentă fezabilă atunci sistemul este robust stabil (i.e., toate traiectoriile rămân mărginite).*

Teorema II.4.10 *Regulatorul DMPC care rezolvă problema de optimizare (II.244) este robust persistent fezabil dacă și numai dacă*

$$\tilde{R}(\tilde{X}_F(\mathbb{X}, N_c, N_p)) \cap \tilde{X}_F(\mathbb{X}, N_c - 1, N_p - 1) \subseteq \tilde{X}_F(\mathbb{X}, N_c, N_p) \quad (\text{II.252})$$

Demonstrație II.4.11 $\tilde{R}(\tilde{X}_F(\mathbb{X}, N_c, N_p))$ este setul de stări atins din setul robust fezabil $\tilde{X}_F(\mathbb{X}, N_c, N_p)$ folosind intrări admisibile, iar setul $\tilde{R}(\tilde{X}_F(\mathbb{X}, N_c, N_p)) \cap \tilde{X}_F(\mathbb{X}, N_c - 1, N_p - 1)$ este subsetul care poate fi atins folosind intrări fezabile care satisfac restricțiile pentru stări. Astfel, după ce se aplică strategia de control fezabilă calculată de regulatorul DMPC, starea la următoarea perioadă de esantionare este $x_i(k + 1|k) \in \tilde{R}(\tilde{X}_F(\mathbb{X}, N_c, N_p)) \cap \tilde{X}_F(\mathbb{X}, N_c - 1, N_p - 1)$. În aceste condiții, dacă $x_i(k + 1|k) \in \tilde{X}_F(\mathbb{X}, N_c, N_p)$, atunci toate evoluțiile viitoare ale sistemului vor rămâne în setul robust fezabil.

II.5 (A2.4) Analiza strategiei de control predictive bazate pe agenți

În continuare se vor analiza strategiile de control DMPC non-cooperativ propuse din punct de vedere al implementării pe un pluton real de vehicule. Astfel, algoritmul DMPC original, dezvoltat pentru un sistem cu doi agenți (i.e., un sistem ce poate fi descompus în două sub-sisteme cuplate) descris în subcapitolul II.2.1 a fost adaptat la o formulare specifică unui pluton de vehicule (i.e., descris printr-o arhitectură în lanț, vezi subcapitolul II.2.1.4).

Peformanțele obținute în simularea unui pluton de vehicule modelate cu dublu integrator arată că introducerea unor restricții pentru variația incrementală a mărimi de comandă, asigură îmbunătățirea performanțelor raportat soluția fără restricții a problemei de optimizare locală.

Această încercare de dezvoltare a unui algoritm predictiv distribuit multi-agent s-a dovedit a fi o testare inițială a conceptului de control al unui pluton de vehicule prin intermediul strategiei DMPC. Deși rezultatele sunt satisfăcătoare, acestea nu garantează aceleași performanțe în implementarea pe un pluton real deoarece modelul folosit în descrierea vehiculelor este extrem de simplist.

A doua strategie propusă (vezi subcapitolul II.2.1.5), folosește același metodologie DMPC aplicată pe un alt model al vehiculului (i.e., model bazat pe descrierea lanțului de transmisie a puterii). Totodată, s-au introdus și perturbații în transmisia semnalelor între vehicule consecutive. Aceste perturbații s-au modelat ca întârzieri de o perioadă de eșantionare între momentul transmisiei traiectoriei optime de la vehiculul predecesor și recepționarea variabilei la vehiculul curent.

Rezultatele obținute arată faptul că efortul de control este din ce în ce mai mare pe măsură ce numărul de vehicule în pluton crește. Înainte, de a fi implementat pe un pluton real, sunt necesare mai multe teste și definirea unor alte soluții de comunicare a informației în pluton, care să fie mai puțin susceptibile la apariția întârzierilor semnalelor (e.g. liderul trimite comanda sa optimă către toate vehiculele următor, rămânând în sarcina agenților locali să păstreze distanța constantă față de liderul plutonului și față de vehiculul predecesor).

După o analiză realizată asupra strategiilor de control descrise pe larg în subcapitolele II.3.3 și II.3.4 pentru plutoanele de vehicule se va determina dacă acestea pot fi aplicate și pentru un pluton real de vehicule.

Plutonul constituit din vehicule care sunt echipate cu sistemul CACC cu reglarea după stare asigură performanțe acceptabile ale plutonului, însă cu cât numărul de vehicule crește cu atât apar perturbații în sistem care se amplifică de-a lungul șirului. Din acest punct de vedere va trebui studiată în amănunt asigurarea stabilității plutonului. Prin faptul că trebuie aplicate referințe de distanță pentru vehiculele controlate acest sistem nu va putea fi aplicat pe un pluton real deoarece în deplasarea lor vehiculele vor urmări doar viteza vehiculului precedent, distanța fiind calculată în funcție de aceasta.

Convoiul de vehicule proiectat cu ajutorul reguletoarelor MPC asigură performanțe superioare față de cazul anterior. În acest caz s-au considerat 15 vehicule în cadrul plutonului,

iar pentru fiecare vehicul urmăritor nu mai este necesară aplicarea unei referințe variabile de distanță pentru a controla distanțele dintre vehicule, ci este suficientă introducerea în sistem a vitezei vehiculului anterior ca și perturbație măsurabilă în funcție de care vehiculele controlate reglează distanțele față de vehiculul precedent. Un impediment al adoptării acestei metode pentru un sistem de timp real ar putea fi reprezentat de volumul mare de calcule pe care trebuie să î-l realizeze regulatorul fiecărui vehicul utilizând funcția *quadprog()* pentru optimizarea problemei pătratice. Este necesară studierea în vederea utilizării altor funcții Matlab (*fmincon()* sau *mpt_solveLP()*) care să permită o execuție mai rapidă a simulării astfel încât după implementarea algoritmului de control pe un pluton real calculele care vor fi efectuate de vehicule să fie suficient de rapide pentru ca acestea să poată reacționa rapid la schimbări imprevizibile de viteză. Și în acest caz este necesară analiza stabilității plutonului.

Etapa III

Implementarea în timp real și analiza performanțelor

III.1 Obiectivele etapei

Obiectiv 3: Implementarea în timp real și analiza performanțelor

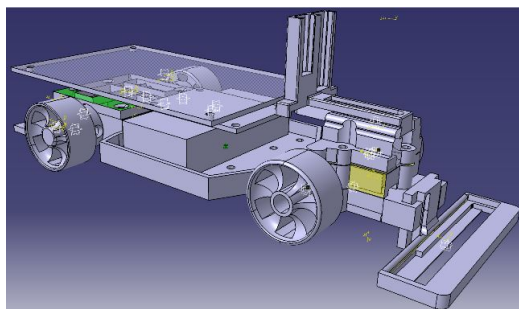
- **Activitate 3.1.** Dezvoltarea programelor de timp real pentru implementarea algoritmilor de control predictivi distribuiți bazați pe agenți în sisteme încorporate
- **Activitate 3.2.** Testarea strategiei de control predictiv distribuit pe un pluton de roboți mobili
- **Activitate 3.3.** Analiza performanțelor în timp real a algoritmilor predictivi bazați pe agenți și compararea cu soluțiile existente pentru plutoanele de vehicule
- **Livrabil 6:** Raport cu descrierea programului de timp real.
- **Livrabil 7:** Raport cu descrierea standului de timp real din laborator și a rezultatelor experimentale cu analiza performanțelor și comparații cu soluțiile existente.

III.2 (A3.1) Dezvoltarea programelor de timp real pentru implementarea algoritmilor de control predictivi distribuiți bazați pe agenți în sisteme încorporate

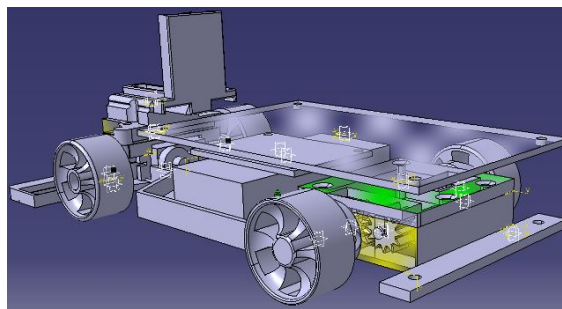
III.2.1 Descrierea mașinii prototip pentru studiul plutoanelor de vehicule

Cercetătorii au abordat studiul plutoanelor de vehicule prin crearea unei platforme care să permită realizarea experimentelor într-un cadru controlat și sigur. Studiul plutoanelor de vehicule cu ajutorul modelelor la scară este de preferat deoarece testele pot fi făcute în interior. Implementările plutoanelor de vehicule bazate pe soluții convenabile ca preț pot face ca în viitor acestea să fie mult mai eficiente și mai rentabile. Experimentele presupun de regulă utilizarea a două sau mai multe vehicule la scară, comandate de câte un microcontroler într-un mediu de laborator controlat și sigur. O implementare convenabilă ca preț dar eficientă a plutoanelor de vehicule a fost proiectată și realizată cu ajutorul a două vehicule la scară, prin utilizarea unor microcontrolere de tip PIC18 și a mai multor tipuri de senzori în [258]. Controlul longitudinal a fost realizat prin utilizarea informațiilor legate de distanță oferite de senzori cu unde infraroșii și cu ultrasunete. Comunicarea între vehicule era bazată pe un modul radio cu frecvența de 434 MHz. O altă abordare este prezentată în [93], unde 3 vehicule robotice cunoscute ca PIE (Platform for Intelligent Embedded Systems) formează un pluton. Comunicarea între vehicule și bază și cea între vehicule se baza pe un modul radio de tip nRF24L01. Controlul traiectoriei precum și al plutonului utilizau informația primită de la o cameră video ce simula prezența unui senzor de tip GPS. O platformă de demonstrație mult mai complexă, compusă dintr-o pistă de curse Carrera și 10 vehicule echipate cu microcontrolere de 32 de biți este prezentată în [94]. Vehiculelor li s-au adăugat componente adiționale, cum ar fi senzori cu infraroșu și module de comunicație radio (xBee 802.15.4 OEM RF). În afară de faptul că pe o pista Carrera vehiculele sunt direcționate automat pe suprafața de rulare, nu a fost necesară implementarea controlului traectoriei. Kesikangas și Sillberg au luat în calcul implementarea și evaluarea controlului longitudinal al plutoanelor de vehicule la scară folosind controlul predictiv și vehicule modificate de la Tamiya [95]. La aceste vehicule, un senzor Hall măsoară viteza de deplasare, o cameră video este folosită pentru a determina distanța și unghiul față de vehiculul precedent, un senzor cu ultrasunete detectează dacă se află vreun vehicul în lateral și un senzor optic cu mai multe celule este folosit pentru implementarea abilității de a urmări o linie de pe suprafața de rulare.

În cadrul proiectului a fost realizat un prototip de vehicul la scară, flexibil și cu opțiuni variate, destinat studiului plutoanelor de vehicule. În comparație cu abordările prezentate, soluția propusă oferă o varietate de senzori și dispozitive ce permit controlul lateral și longitudinal al plutonului de vehicule, comunicație radio și o extindere ușoară a sistemului încorporat. Vehiculul permite înregistrarea și stocarea a diferiți parametri în fișiere specifice pe SD card, permițând realizarea unei analize offline mult mai ușoare. Sistemul încorporat este modular și este format dintr-o placă principală la care sunt conectați toți senzorii și modulele, și o placă secundară, opțională, ce poate fi folosită pentru implementarea unor



(a) Vedere laterală față



(b) Vedere laterală spate

Figura III.1: Vederi de tip 3D ale vehiculului prototip

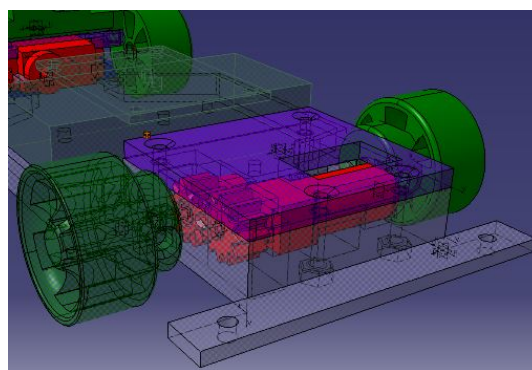
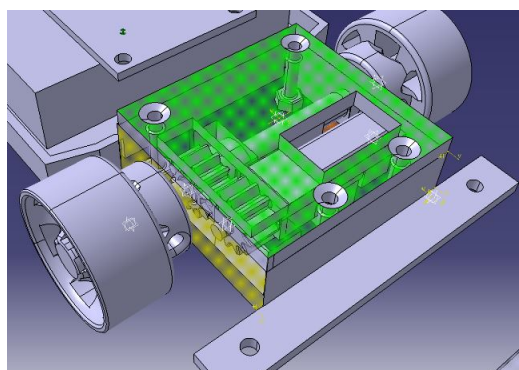


Figura III.2: Vederi de tip 3D ale tracțiunii mașinii prototip

algoritmi de control ce necesită putere de calcul mărită. Una dintre principalele cerințe pentru mașina prototip a fost posibilitatea de a face experimente cu până la 10 vehicule pentru a observa mai bine comportamentul plutonului. S-a dorit ca toate vehiculele să fie identice, să semene cu un vehicul real, să fie capabile să urmărească o linie de pe suprafața de rulare, să aibă spațiul necesar pentru montarea tuturor senzorilor, să fie capabile să salveze date într-un anumit format pe un SD CARD și să aibă încorporat un modul pentru comunicație radio. În Fig. III.1 sunt ilustrate vederile laterale față de tip 3D ale vehiculului prototip.

În locul achiziției unor modele la scară din comerț și a modificării acestora pentru a instala pe ele senzorii și modulele dorite, soluția aleasă a constat în proiectarea și croirea mașinii prototip la scară într-o aplicație de tip CAD 3D și realizarea părților componente folosind o imprimantă 3D. Prototipul a fost proiectat astfel încât să semene cu un vehicul real. Direcția acționează asupra roților din față iar tracțiunea este implementată cu ajutorul unui motor de curent continuu cu perii montat în partea din spate. Motorul destinat propulsiei are atașat un encoder magnetic ce permite determinarea facilă a vitezei mașinii prototip. Direcția respectă geometria Ackermann și este controlată de un servomotor. Ilustrări grafice 3D ale părții din spate a prototipului, inclusiv partea de propulsie, sunt prezentate în Fig. III.2.

În locul folosirii unei camere video sau a unui modul GPS pentru controlul traiectoriei, vehiculele din pluton se vor comporta ca niște urmăritori de linie. O linie închisă la culoare pe o suprafață deschisă la culoare indică traectoria pe care vehiculele trebuie să o urmeze. Fiecare vehicul este echipat cu un senzor de linie cu infraroșu ce transmite poziția liniei

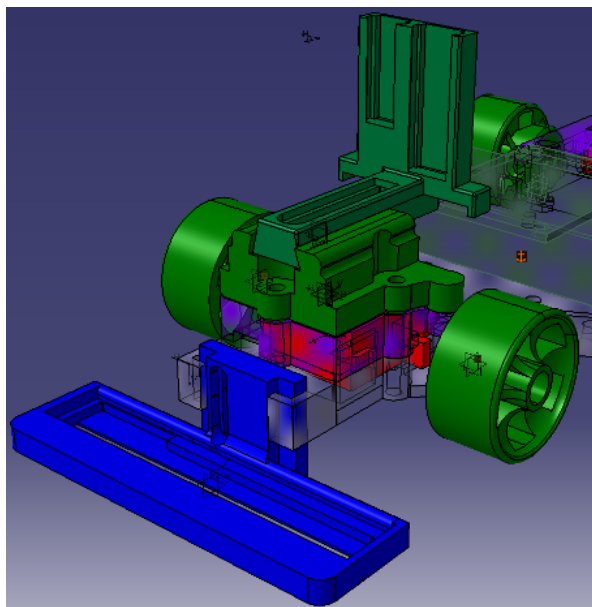


Figura III.3: Vedere de tip 3D din față a mașinii prototip.

către microcontroler. Această informație este folosită de către algoritmul de reglare al traiectoriei pentru a comanda servomotorul și a implementa traiectoria dorită. Prototipul folosește doi senzori de distanță. Primul este un senzor de distanță cu infraroșu ce va transmite informații legate de distanța față de mașina din față. Această informație este folosită de către algoritmul de control al distanței pentru a comanda motor de propulsie. Al doilea senzor de distanță este un senzor cu infraroșu de proximitate și este folosit pentru a detecta dacă vehiculul din față se află prea aproape, caz în care va declanșa o întrerupere, astfel încât vehiculul va fi oprit imediat, fiind astfel evitată coliziunea. Ambii senzori sunt montați pe un suport mobil astfel încât distanța minimă de evitare a coliziunii poate fi modificată foarte ușor. O vedere 3D din față a prototipului, ce prezintă suportul senzorului de linie, suportul senzorilor de distanță precum și geometria direcției este prezentată în Fig. III.3.

Fiecare vehicul are un accelerometru și o busolă ce returnează informații utile privind orientarea mașinii și accelerațiile statice și dinamice la care este supus vehiculul. Accelerometrul permite sistemului încorporat să "înțeleagă" mediul înconjurător și să realizeze un control mai bun al vitezei. Poate fi folosit împreună cu encoderul motorului de curent continuu pentru a realiza algoritmi de start și stop precum și alte funcționalități utile, cum ar fi detectarea lovirii sau ridicării vehiculului. Vehiculele din pluton sunt capabile să schimbe informații prin intermediul unui modul radio. Fiecare modul radio poate funcționa atât ca transmițător, cât și ca receptor, astfel încât diferite arhitecturi de comunicație între vehicule pot fi implementate și testate.

Una dintre cele mai importante cerințe legate de design-ul vehiculului a fost să fie capabil să salveze diverse informații legate de activitatea sa în fișiere specifice, în loc să le trimită la un calculator prin comunicația radio. În acest fel este disponibilă o lățime mai mare de bandă pentru comunicația dintre vehicule iar fișierele cu date pot fi încărcate pe un

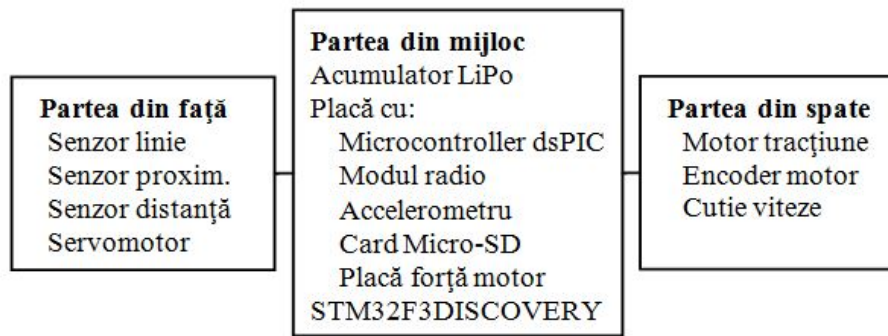


Figura III.4: Poziționarea modulelor componente pe vehiculul prototip.

calculator și analizate ulterior. Pentru aceasta este folosit card SD. O baterie reîncărcabilă de tip LiPo cu 2 celule și capacitate de 1300 mA alimentează vehiculul. Alegerea a fost determinată de către caracteristicile bateriilor LiPo: greutate redusă, curent mare de ieșire și rată de descărcare scăzută. În Fig. III.4 sunt prezentate toate modulele din care este format prototipul, precum și poziționarea acestora.

III.2.1.1 Proiectarea și realizarea părții mecanice a mașinii prototip

În vederea realizării tuturor componentelor mecanice ale mașinii prototip a fost necesară îndeplinirea următoarelor obiective:

- realizarea componentelor unui vehicul prototip în mediul virtual Catia;
- asamblarea virtuală a componentelor vehiculului;
- simularea mișcării cinematice pentru articulațiile mobile;
- realizarea componentelor la imprimantă 3D fără necesitatea ulterioară de a aduce modificări pieselor pentru a se potrivi.

În vederea realizării unei piese, de la stadiul de proiectare până la obținerea produsului finit, au fost parcurși o serie de pași.

- După formarea cerințelor de proiectare, piesa a fost proiectată în mediul CAD (computer-aided design) cu o soluție software specifică mediului de proiectare 3D. Soluția aleasă a fost folosirea aplicației Catia, unul dintre cele mai folosite software-uri din domeniul proiectării CAD. Catia permite proiectarea obiectelor 3D (în coordonatele x, y, z), vizualizarea spațială a acestora (Fig. III.5), asamblarea virtuală a obiectelor formate din mai multe piese precum și simularea mișcării componentelor mobile.
- Odată proiectarea piesei realizată, proiectul este salvat în unul din formatele obiect solid (de obicei. STL) și este prelucrat de o aplicație ce transformă acest format în limbaj cod mașină. Acest limbaj este specific imprimantelor 3D, numele formatului este G-code (fișierele având extensia .Gcode) și conținutul fișierului reprezintă toate coordonatele pe care capul de imprimare (extruderul) trebuie să le parcurgă, precum

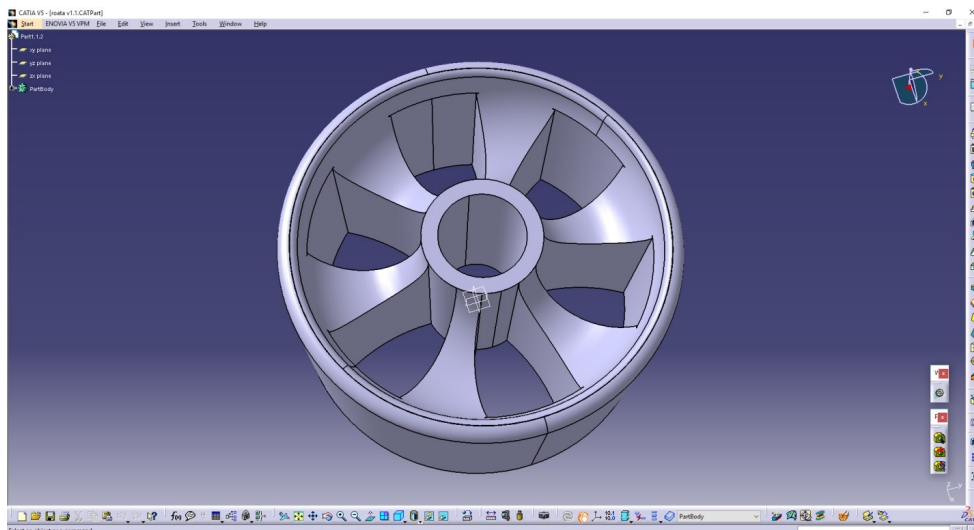


Figura III.5: Proiectarea obiectelor 3D în Catia.

și temperaturi necesare și momentele la care se dorește intervenția răcirii. Aplicația folosită pentru crearea fișierelor G-code se numește Slic3r (aplicație gratuită, de tip open source) și prin intermediul ei se pot seta viteza de imprimare, modul de imprimare precum și alte facilități ce țin de realizarea obiectului în formă finită. Principiul de generare a codului constă în împărțirea înălțimii piesei în straturi (în acest caz 0,25 mm) după care se generează traseul/perimetrul ce va fi urmat de capul de imprimare. În Fig. III.6 este reprezentat un singur stat ce dorim să fie imprimat din obiectul reprezentat în Fig. III.5. Fișierul G-code conține coordonatele în planele x, y, z, ce urmează a fi parcurse de capul de imprimare, temperaturile suportului de imprimare și capului de imprimare în diferite momente ale imprimării, intervențiile ventilatorului și alte informații privind imprimarea.

- Fișierul G-code poate fi importat în aplicația Repetier-Host, ce permite accesul la G-code, vizualizarea și editarea traseului pe care îl va urmări imprimanta. Acest pas este opțional, fișierul G-code putând fi încărcat pe imprimanta 3D prin intermediul unui card SD (cea mai folosită soluție) sau transmis de pe calculator prin interfața USB. În unele cazuri fișierele G-code generate de Slic3r au mici erori ce produc decalaje între straturi, imprimare incorectă sau chiar oprirea imprimantei. Aceste erori se produc, de obicei, atunci când se fac mutări pe platan (în Slic3r), rotiri sau scalări ale piesei. Aceste erori pot fi observate și remediate cu ajutorul programului Repetier-Host ce are o interfață interactivă, ce permite vizualizarea imprimării în fiecare punct pe care îl parcurge capul de imprimare. În Fig. III.7 este prezentat modul de editare a fișierului G-code produs de Slic3r, pentru realizarea unei anumite părți din piesa finită. O altă facilitate oferită de Repetier-Host este aceea de a șterge partea de început a unei piesei, lucru foarte util pentru a continua imprimarea după o eventuală oprire neprogramată.

Pașii prezentați mai sus sunt ilustrați în Fig. III.8 și în Fig. III.9.

Cea mai importantă piesă a mașinii prototip este șasiul, acesta reprezentând platforma pe

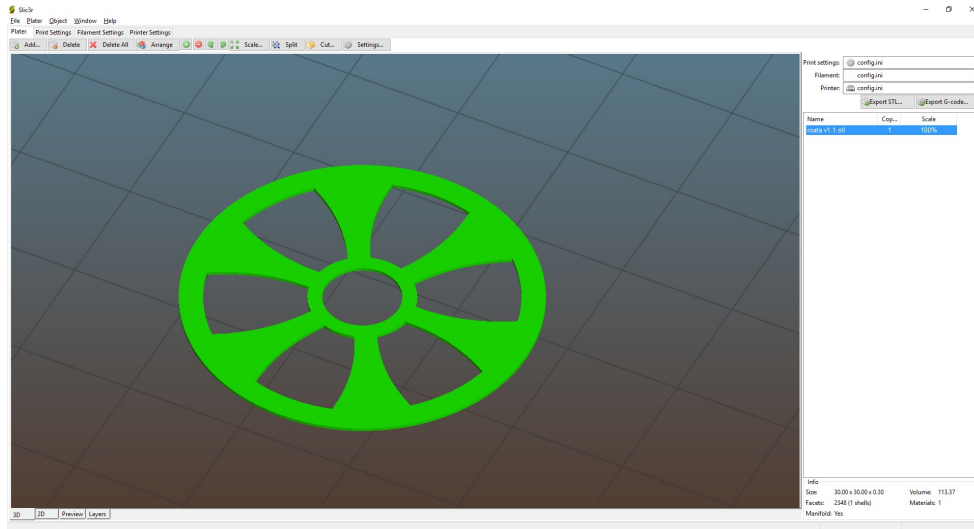


Figura III.6: Transformarea obiectului în G-code cu Slic3r.

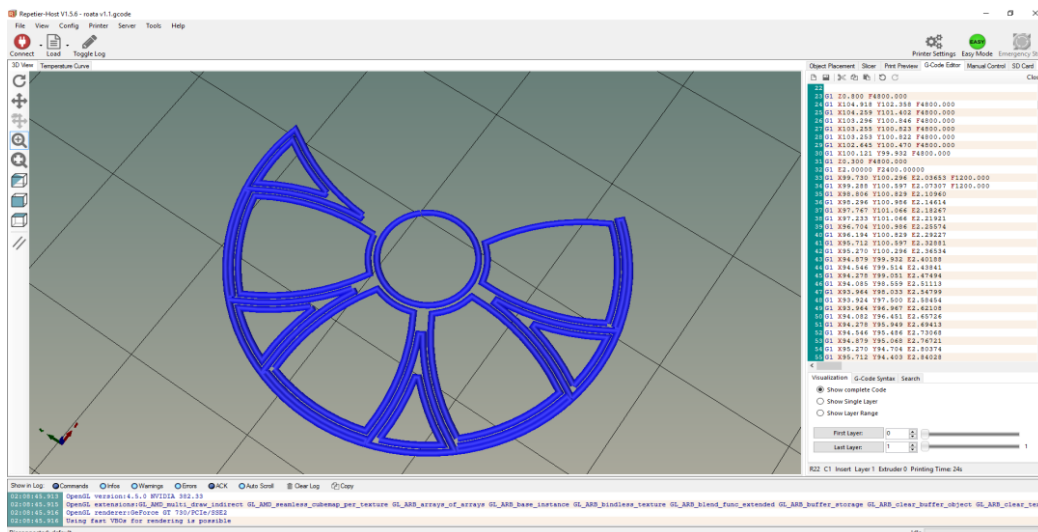


Figura III.7: Editarea fișierului G-code în Repetier-Host.

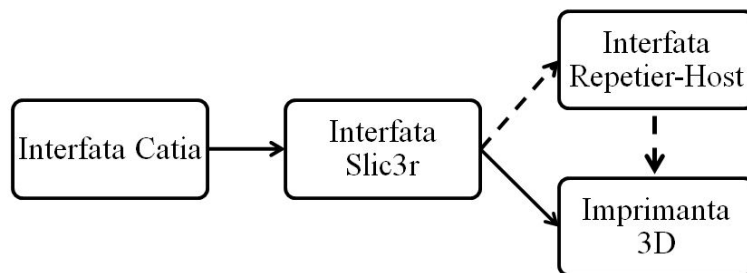
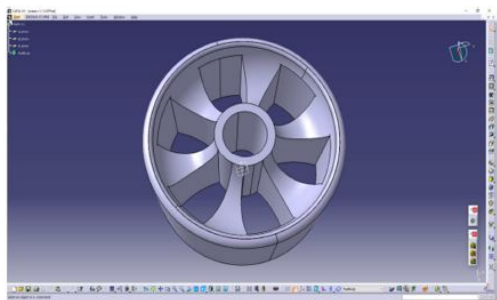
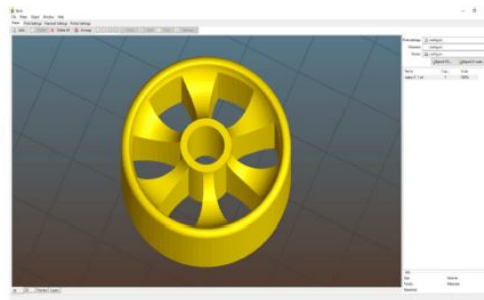


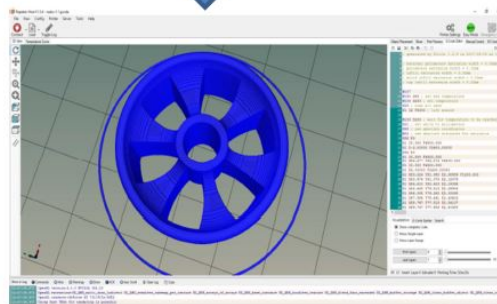
Figura III.8: Etapele necesare realizării unui prototip de piesă.



Realizarea obiect in CAD



Transformare obiect in format G-code



Editarea codului in Repetier-Host



Rezultatul final

Figura III.9: Etapele necesare realizării unui prototip de piesă.

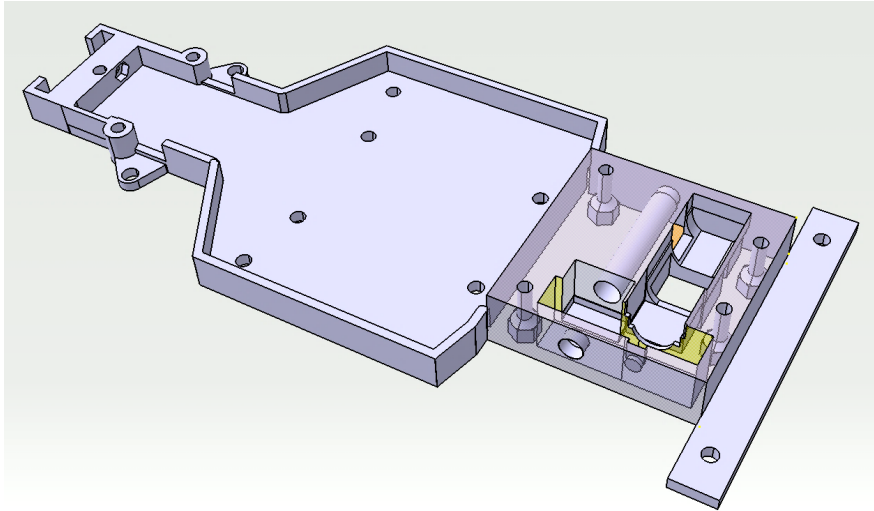


Figura III.10: asiul proiectat.

care au fost montate celelalte piese și module. Proiectarea acestuia a fost împărțită în 4 zone:

- zona în care a fost montat motorul de propulsie, zonă caracterizată de următoarele caracteristici: poziționare fixă și optimă a motorului, amplasarea optimă a axului propulsor și reducerea frecărilor acestuia, zona de răcire a motorului, zona de protecție împotriva prafului a ansamblului de pinioane;
- zona alocată direcției Ackermann, cu următoarele caracteristici și cerințe de proiectare: fixare optimă și fixă a servomotorului, prevedere traseu pentru conexiunile electrice, găuri piesa prindere servomotor și suport senzori distanță, găuri prindere bielete pe sașiu;
- partea de rezistență, cu următoarele caracteristici și detalii importante: găuri destinate montării bateriei, găuri destinate prinderii plăcii principale cu microcontroller, zonă mediană sașiu întărită și rigidizată;
- zona de fixare a suportului sensorului de linie, ce permite o reglare facilă a înălțimii sensorului de linie.

Fig. III.10 și Fig. III.11 prezintă șasiul proiectat în Catia respectiv șasiul realizat la imprimanta 3D.

Similar, Fig. III.12, Fig. III.13 și Fig. III.14 sunt prezentate toate piesele componente ale mașinii prototip (piesa proiectată respectiv piesa printată 3D).

Pentru transmiterea mișcării de rotație de la motorul propulsor la axul propulsor, soluția aleasă a constat în folosirea a 2 roți dințate (pinioane), egale în diametru, raportul de transmisie a puterii fiind de 1:1. Pentru realizarea pinioanelor, cerințele de proiectare au fost:

- roțile dințate să nu se blocheze în timpul mișcării de rotație;
- raportul de transmisie a puterii să fie de 1:1;

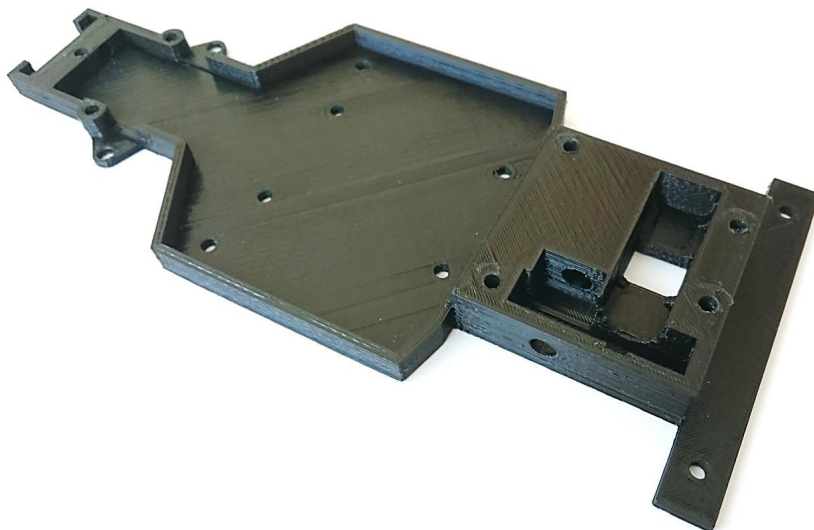


Figura III.11: asiul imprimat 3D.

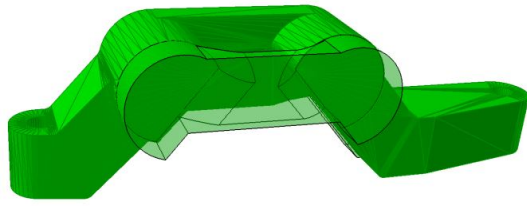
- să se permită un joc limitat între pinioane, pentru compensarea eventualelor mișcări de rotație neliniare ale motorului;
- diametrul pinionului să nu depășească 14 mm;
- roțile dințate să fie protejate împotriva intrării de particule, apă și praf;
- roțile dințate să poată fi incluse în zona special destinată din șasiu.

Roata dințată activă (sau pinionul principal) este roata montată pe axul motorului propulsor, transmițând cuplul către roata pasivă (pinionul secundar) ce este montată pe axul propulsor. Cerințele de proiectare au fost următoarele:

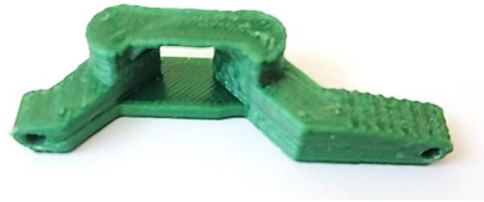
- potrivirea pinionului pe axul motorului (diametru 3mm, teșit în formă de D);
- rezistență mărită în zona de contact dintre axul motorului și pinion;
- suprafață teșită ce reprezintă primul strat de imprimare;
- diametrul exterior al pinionului să nu depășească 14 milimetri;
- numărul de dinți să fie 12.

Roata dințată pasivă (sau pinionul secundar) este pinionul montat pe axul propulsor, căruia i se transmite cuplul generat de motorul propulsor. Cerințele de proiectare ale pinionului secundar au fost următoarele:

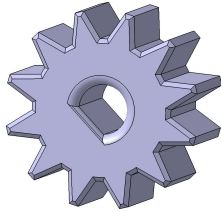
- potrivirea centrului pinionului cu centrul axului propulsor;
- fixarea pinionului pe axul propulsor cu șurub cu diametrul de 2 milimetri;
- locaș pentru introducerea piuliței ce fixează șurubul;
- șurubul să treacă prin axul propulsor;



(a) Cadru direcție - proiectat.



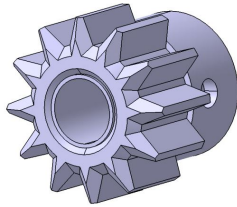
(b) Cadru direcție - printat 3D.



(c) Pinion principal - proiectat.



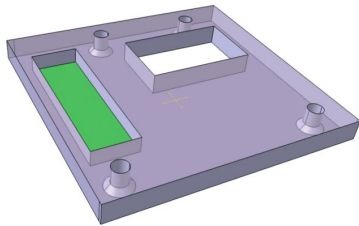
(d) Pinion principal - printat 3D.



(e) Pinion secundar - proiectat.



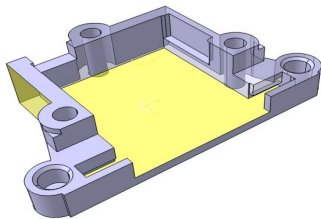
(f) Pinion secundar - printat 3D.



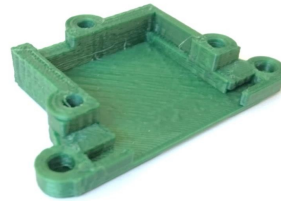
(g) Placă prindere motor - proiectată.



(h) Placă prindere motor - printată 3D.



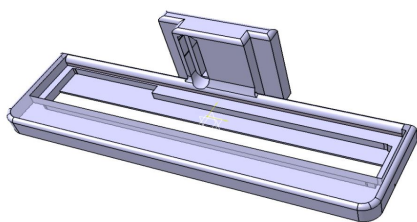
(i) Capac servomotor - proiectat.



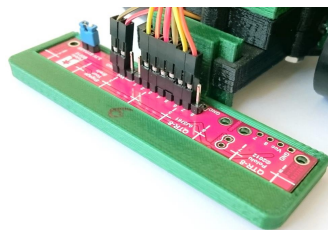
(j) Capac servomotor - printat 3D.

Figura III.12: Piesele proiectate în Catia și respectiv printate 3D - partea I

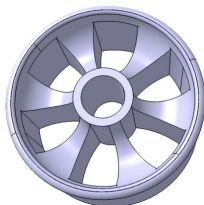
- rezistență ridicată în zona prinderii pe axul propulsor;
- rezistență ridicată a dinților;
- diametrul pinionului să nu depășească 14 milimetri;



(a) Suport senzor de linie - proiectat



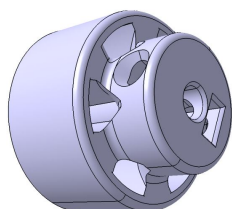
(b) Suport senzor de linie - printat 3D



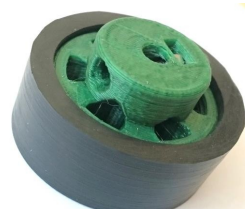
(c) Roată față - proiectată



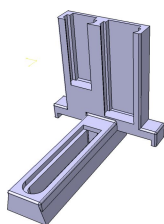
(d) Roată față - printată 3D



(e) Roată spate - proiectată



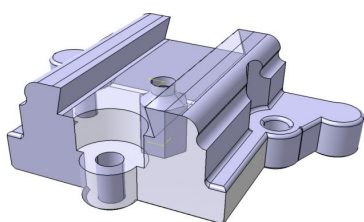
(f) Roată spate - printată 3D



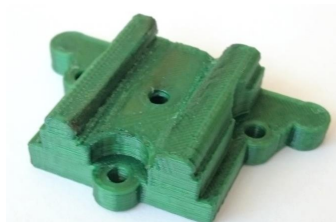
(g) Suport senzori infraroșu - proiectat



(h) Suport senzori infraroșu - printat 3D



(i) Bază suport senzori infraroșu - proiectat



(j) Bază suport senzori infraroșu - printat 3D

Figura III.13: Piesele proiectate în Catia și respectiv printate 3D - partea a II-a

- diametrul prinderii să nu depășească 9 milimetri;
- margine teșită pe partea pe care se imprimă piesa.

Cerințele de proiectare ale suportului senzorului de linie au fost următoarele:

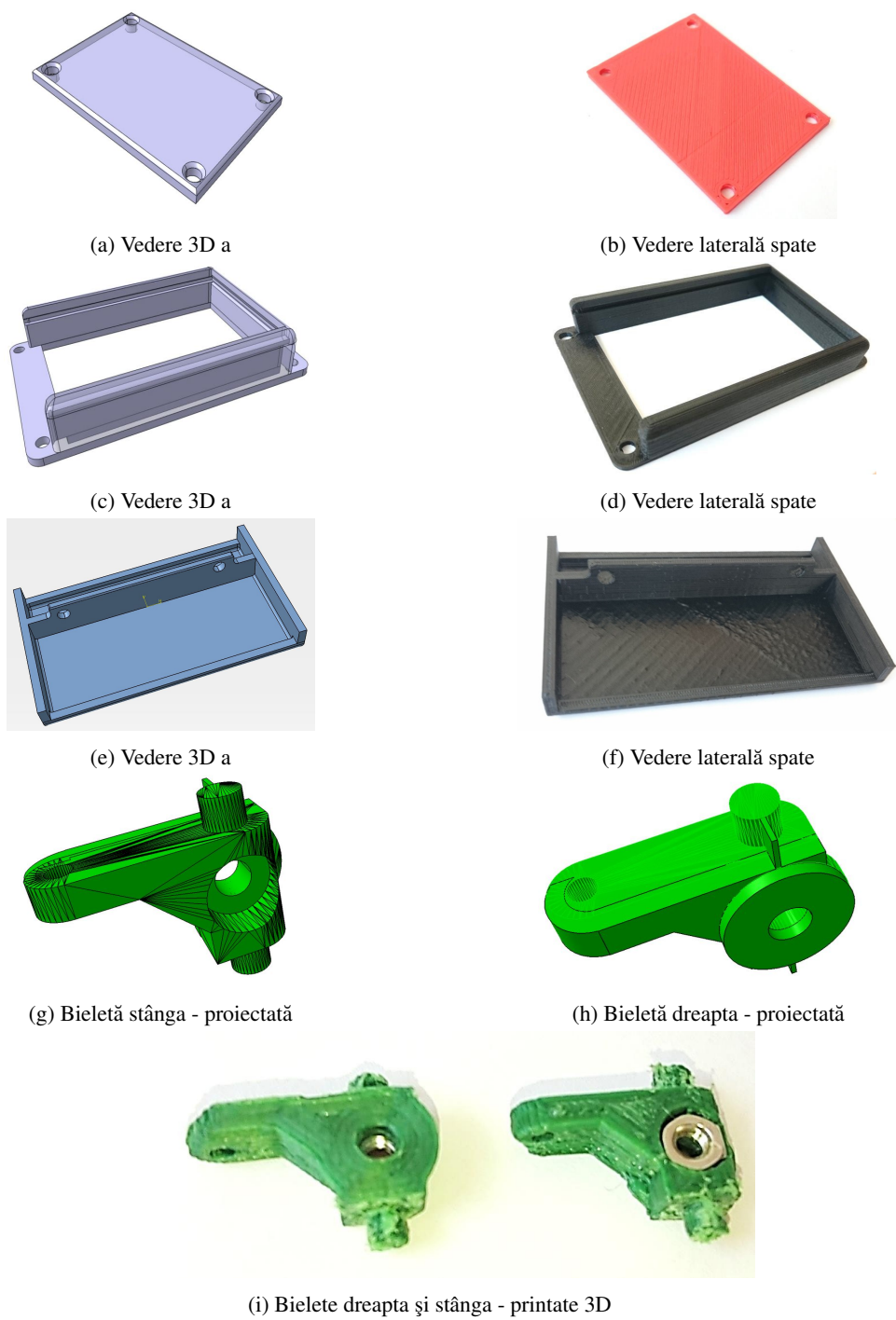


Figura III.14: Piesele proiectate în Catia și respectiv printate 3D - partea a III-a

- suportul să poată fi reglat pe înălțime;
- suportul să poată fi fixat pe o poziție dorită;

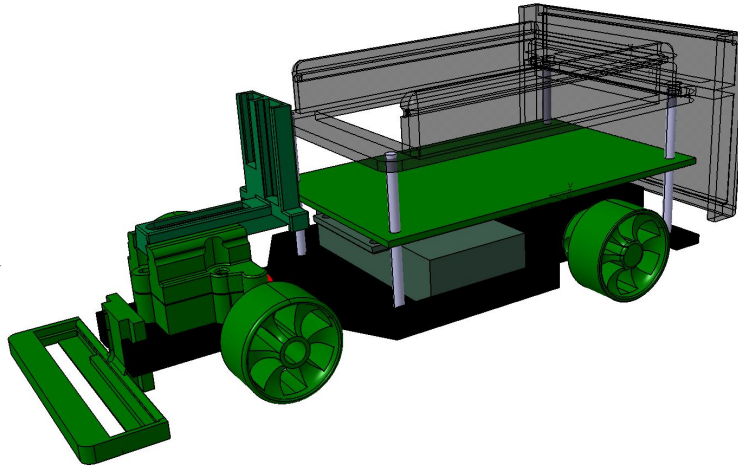


Figura III.15: Mașina prototip proiectată în Catia cu toate piesele componente asamblate.

- suportul trebuie să protejeze senzorul fără să îi afecteze funcționalitatea;
- suportul să aibă rolul de a proteja și devia vehiculul în caz de coliziune;
- să se potrivească pe cadrul șasiului;
- suprafață teșită pe care se face imprimarea;
- să fie suficient de rezistent pentru a rezista la coliziuni;
- partea inferioară să acopere o suprafață mai mare pentru a reduce influența luminii;
- senzorul pentru care se face suportul este QTR-8A de la Pololu.

Fig. III.15 și Fig. III.16 prezintă mașina prototip proiectată în Catia cu toate piesele componente asamblate respectiv mașina prototip realizată real.

III.2.1.2 Prezentarea senzorilor și dispozitivelor ce compun sistemul încorporat al mașinii prototip

Sistemul încorporat este unul modular și permite o configurare flexibilă. Placa principală a fost proiectată astfel încât să poată fi conectați toți senzorii și toate modulele prezentate în secțiunile anterioare. Placa conține un microcontroller dsPIC ce se poate ocupa de toate funcționalitățile de bază, cum ar fi citirea informațiilor de la senzori, comunicația folosind modulul radio, salvarea datelor pe card-ul SD, implementarea controlului traiectoriei și implementarea algoritmilor de control cu cerințe reduse privind puterea de calcul. Modulul NRF24L01 (Fig. III.17) reprezintă un modul de comunicație wireless, în banda 2.4GHz ISM (Industrial, Scientific and Medical). Modulul este de putere mică, tensiunea de alimentare fiind între 1.9 și 3.6 Volți, iar curentul în timpul transmisiei și al recepției nu depășește 14mA. Modulul comunică cu microcontrolerul prin interfața SPI. Viteza maximă de comunicație posibilă este de 2 Mbps. Dacă se dorește extinderea distanței de transmisie se poate alege o viteză mai mică. Lungimea pachetului de date este de 32 de octeți iar

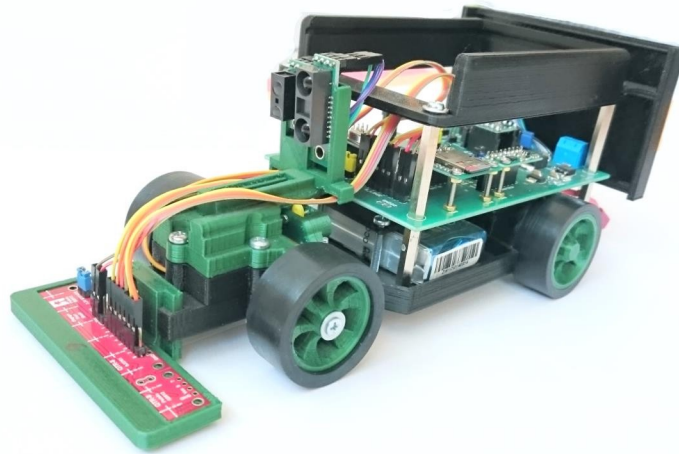


Figura III.16: Mașina prototip realizată real.



Figura III.17: Modulul radio nRF24L01.

adresa este pe 5 octeți. Un mare avantaj al acestor module constă în faptul că au implementat un cod corector de erori, pe 8 biți, făcând comunicația sigură. Distanța maximă testată la care s-a reușit comunicația fără a apărea întârzieri majore în recepția mesajelor este în aer liber de 100 de metri.

Adaptorul pentru SD card este de tip POLOLU-2587. Este compact și oferă acces la pinii necesari unui card microSD pentru a comunica prin SPI, putând fi integrat direct în sistemele de 5V deoarece are un regulator de 3.3 V integrat. A fost ales datorită mărimii și a conecticii ușor de realizat. Adaptorul poate fi folosit doar în modul SPI. Se alimentează la 5V prin intermediul pinului VDD iar toți pinii logici pot fi conectați la microcontrolere de 5V datorită adaptoarelor de nivel incluse pe placă. Regulatorul 3V3 poate fi oprit prin intermediul pinului EN pentru un consum redus atunci când cardul nu este folosit. În Fig. III.18 este prezentată configurația pinilor adaptorului.

Un alt modul utilizat în cadrul sistemului încorporat al mașinii prototip este LSM303D. Acesta combină un accelerometru digital pe 3 axe și o busolă (senzor magnetic). Senzorul returnează șase valori ce pot fi citite independent și a căror sensibilitate poate fi reglată

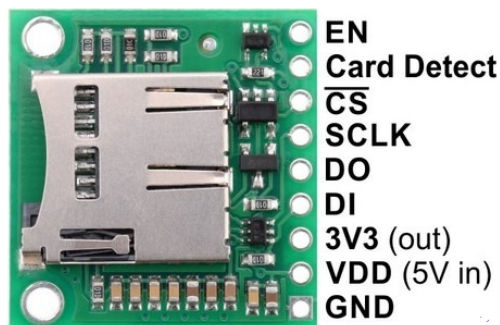


Figura III.18: Adaptorul pentru microSD card POLOLU-2587.

între 2 și 16 g pentru accelerometru și 2 până la 12 gaus pentru senzorul magnetic. Valorile pot fi citite prin interfața I2C sau SPI. Placa cu senzorul LSM303D conține un regulator de tensiune de 3.3V și adaptoare de nivel ce fac posibilă folosirea senzorului cu tensiuni între 2.5V și 5.5V. În Fig. III.19 este prezentată o vedere de sus cu descrierea pinilor. LSM303D are multe opțiuni configurabile cum ar fi sensibilitate variabilă configurabilă dinamic, posibilitatea alegerii intervalului de valori pentru ieșire și doi pini pe care se pot genera întreruperi externe către microcontroller. Accelerometrul și senzorul magnetic pot fi porniți independent în funcție de necesități pentru a reduce consumul de energie. Inițial senzorul comunică prin I2C. Pentru aceasta sunt necesari 2 pini logici (SCL și SDA). Acești pini sunt conectați la adaptoarele de nivel și pot fi folosiți cu tensiuni mai mari de 3.3V. Trebuie conectați la o magistrală I2C ce funcționează la același nivel logic cu VIN. Pentru a folosi senzorul în modul SPI este nevoie de 4 pini în mod uzual (SPC, SDI, SDO și CS). Aceștia trebuie de asemenea conectați la o magistrală SPI ce funcționează la același nivel logic cu VIN. Interfața SPI poate fi setată să funcționeze și cu 3 pini (SDO și SDI pe același pin). Atunci când pinul CS este în starea lui inițială (conecat la VDD), LSM303D comunică prin magistrala I2C. Adresa senzorului în modul I2C este 0011101b. Ultimii doi biți sunt dați de tensiunea pe pinul SA0. Dacă un alt dispozitiv de pe magistrală are aceeași adresă, prin conectarea pinului SA0 la masă adresa va deveni 0011110b. Interfața I2C de pe senzor are o viteză maximă de 400 kHz (I2C fast mode standard). Pentru a trece senzorul în modul SPI, pinul CS trebuie pus la masă înainte de a trimite orice comandă pe SPI și apoi din nou la VDD la sfârșitul comenzii.

Microcontrollerul din placa principală a sistemului încorporat comanda motorul de tracțiune prin intermediul plăcii de forță MAX14870. Acest modul folosește driverul pentru motoare MAX14870 și oferă o gamă largă de tensiuni de alimentare (între 4.5V și 36V). Curentul maxim ce poate fi furnizat continuu este de 1.7A pentru un singur motor cu perii. Are câte un pin pentru controlul direcției și al vitezei, protecție la supratensiune, la subtensiune, la supraîncălzire și la tensiune inversă. Driverul MAX14870 este o punte H ce reprezintă o alegere bună pentru un singur motor DC cu perii de dimensiuni reduse. Faptul că poate furniza un curent continuu de 1.7 A și timp de câteva secunde 2.5 A prezintă un avantaj pentru sistemul încorporat atașat mașinii prototip. În Fig. III.20 este prezentat un exemplu de conectare minimală pentru a putea folosi driver-ul. Pinii pentru alimentare și conectarea

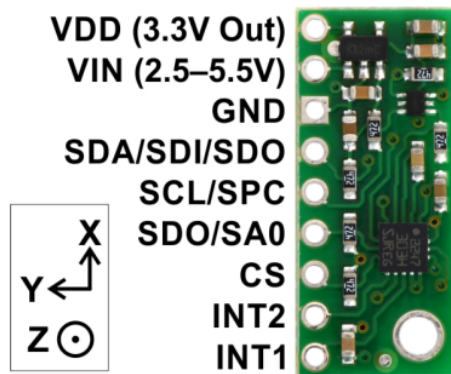


Figura III.19: Senzorul accelerometru și busolă de tip LSM303D.

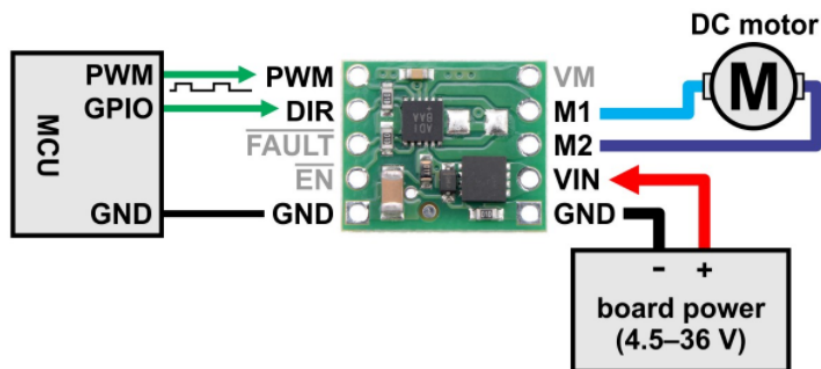


Figura III.20: Placa de forță cu MAX14870.

motorului sunt de o parte a plăcii iar cei pentru control de cealaltă parte. Pin-ul VIN se conectează la sursa de alimentare între 4.5V și 36V, iar pinul VM oferă acces la tensiunea de intrare VIN protejată contra tensiunii inverse. Controlul se realizează simplu prin intermediul a doi pini (DIR și PWM). Pin-ul DIR determină direcția motorului iar pinul PWM este folosit pentru a controla viteza de rotație a motorului. Inițial pinul PWM este tras la masă prin intermediul unei rezistențe, ceea ce înseamnă că ambele ieșiri către motor sunt de asemenea la masă rezultând frânarea motorului. Pinul EN poate fi folosit pentru a deconecta ieșirile către motor, acest lucru fiind folosit dacă se dorește lăsarea motorului să se învârtă liber. Inițial pinul este pus la masă, ceea ce înseamnă că motorul este conectat la driver. Driver-ul mai are o funcție de limitare de curent ce poate fi activată prin lipirea unei rezistențe pe placă, în funcție de valoarea maximă a curentului dorită. Dacă se depășește limitarea de 2.5A driver-ul se va încălzi excesiv iar protecția la temperatură va deconecta motorul.

Motorul este unul de curent continuu cu perii conectat la un reductor din metal cu un raport de transmisie de 100:1. Alimentat la o tensiune de 6V se va roti cu 320 RPM și va consuma 120 mA fără sarcină și 1.6A atunci când este blocat, ceea ce îl face perfect pentru a fi folosit împreună cu driver-ul de motoare MAX14870 (Fig. III.21).

Pentru a determina viteza motorului se poate atașa motorului un senzor magnetic (Fig.



Figura III.21: Motorul de tracțiune.

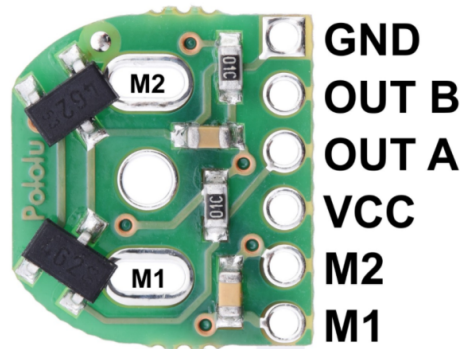


Figura III.22: Encoderul magnetic.

III.22). Acesta oferă o rezoluție de 12 impulsuri pe o rotație a axului motorului. Senzorul funcționează cu o tensiune între 2,7V și 18V și oferă ieșiri digitale ce pot fi conectate direct la un microcontroler sau la un alt circuit digital. Senzorul include doi senzori cu efect Hall cu două canale și un disc magnetic cu 6 poli ce poate fi folosit pentru codificarea în quadratură. Senzorul situat pe placă detectează rotația discului magnetic și oferă 12 impulsuri pe rotație atunci când se iau în calcul fronturile de pe ambele canale. Pentru a afla numărul de impulsuri pe rotație ale reductorului se înmulțește rata de multiplicare a reductorului cu 12. Senzorul este alimentat prin intermediul pinilor VCC și GND. Ieșirile în quadratură A și B sunt semnale digitale conectate la masa sau la VCC prin intermediul rezistorilor pull-up în funcție de câmpul magnetic. Distanța dintre discul encoderului și placa cu senzori nu are un impact mare în performanța acestora atât timp cât axul motorului trece prin tot discul.

Senzorul de linie este de tip QTR-8C. Are 8 LED-uri cu infraroșu și 8 fotorezistori făcându-l folosit pentru detecția unei linii (Fig. III.23). Perechile de LED-uri sunt conectate în serie pentru a înjumătăți consumul de curent și pot fi oprite de tot pentru un consum și mai redus sau pentru a modifica intensitatea lor prin PWM prin intermediul unui tranzistor de tip MOSFET. Tranzistorul are baza conectată la VCC în mod normal, LED-urile fiind pornite inițial. Prin conectarea la masă a bazei tranzistorului LED-urile sunt oprite. Fiecare sen-

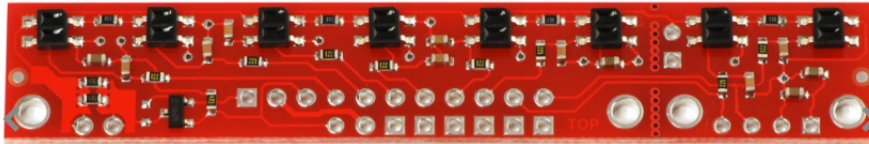


Figura III.23: Senzorul de linie de tip QTR-8C.

zor oferă o ieșire digitală separată. Senzorul funcționează corect dacă placa este paralelă cu suprafața pe care se dorește detecția liniei. Distanța optimă de măsurare este de 3mm. Pentru a folosi senzorul trebuie aplicată tensiune pe pin-ul de ieșire al microcontrolerului conectat la senzor. Apoi reflectanța se determină măsurând timpul necesar pentru ca tensiunea să scadă către 0. Un timp mai scurt de scădere înseamnă o reflectanță mai bună. Rezultate valabile se pot citi după o milisecundă. O secvență tipică pentru a citi un senzor este următoarea:

- Se pornesc LED-urile (opțional, doar dacă au fost oprite);
- Se setează pin-ul corespunzător senzorului ca ieșire și se aplică tensiune pe el;
- Se așteaptă cel puțin 10 s pentru ca tensiunea de ieșire a senzorului să crească;
- Se reconfigurează pin-ul ca intrare digitală;
- Se măsoară timpul până când scade tensiunea (starea pin-ului este low);
- Se opresc LED-urile (opțional).

Acest mod de măsură aduce multe avantaje, cum ar fi faptul că nu sunt necesare convertoare analog-digitale, se pot citi mai multe intrări în paralel cu majoritatea microcontrolerelor și se poate reduce consumul prin oprirea LED-urilor. Consumul de curent al unui LED este între 20 mA și 25 mA, făcând ca consumul total al plăcii să fie sub 100mA.

Senzorul de obstacol POLOLU-1134 (Fig. III.24) poate detecta obiecte la o distanță cuprinsă între 2 cm și 10 cm. Este un senzor cu răspuns rapid, de dimensiuni reduse și cu consum minim de curent. Este excelent pentru detecția obstacolelor datorită intervalului de detecție. Senzorul doar specifică dacă este sau nu un obiect în raza acestuia de detecție, distanța față de acesta nefiind precizată. Funcționează cu o tensiune de alimentare cuprinsă între 2.7 V și 6.2 V. Pe spatele plăcii cu senzorul se află și un LED ce semnalizează prezența unui obstacol. Cu LED-ul conectat, pe pin-ul de ieșire tensiunea atunci când este detectat un obiect scade la 1V, iar cu LED-ul deconectat va fi de 0,96V (alimentat la 5V) sau de 0,46V (alimentat la 3.3V). Senzorul mai are și un pin de "Enable" ce poate fi folosit pentru a trece senzorul în modul consum redus, acesta fiind conectat inițial la Vcc, senzorul fiind în permanență activ.

Senzorul de distanță POLULU 2476 (Fig. III.25) poate detecta și măsura distanța pentru obiecte aflate la o distanță între 10 cm și 150 cm. Distanța este indicată prin intermediul unei tensiuni analogice, deci se poate folosi o singură intrare analogică pentru detecție. Pin-ul de ieșire analogică OUT trebuie conectat la o intrare analogică a microcontrolerului. Senzorul actualizează automat ieșirea la fiecare 16 mS. Pinul EN poate fi pus la masă

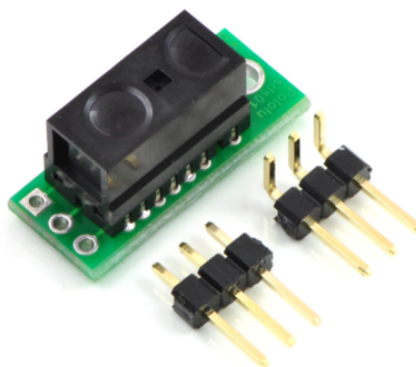


Figura III.24: Senzorul de proximitate POLULU-1134.

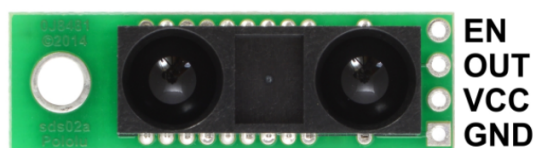


Figura III.25: Senzorul de distanță de tip POLOLU-2476.

pentru a trece senzorul în modul de consum redus. Acest pin este conectat la VCC inițial prin intermediul unei rezistențe de pull-up. Senzorul este setat în modul de funcționare 3V, tensiunea de alimentare necesară fiind între 2.7V și 3.6V, iar diferența de tensiune pe intervalul de distanță măsurat este de 1.6V (1.9V la 10 cm, 0.3V la 150 cm).

Servomotorul HD-1900A (Fig. III.26) poate fi folosit ca și actuator pentru mecanismele fine. Pentru a controla poziția servomotorului este necesar un semnal PWM cu perioada de 20 ms (50 Hz) și lățime a impulsului între 1ms și 2 ms.

Pentru alimentarea celor patru motoare și a restului componentelor este folosit un acumulator LiPo (Litiu-Polimer) cu 2 celule conectate în serie. Fiecare celulă are o tensiune de 3.7V. Capacitatea maximă a acumulatorului este de 1300mAh (miliamperi pe oră), adică poate furniza un curent de 1,3 Amperi timp de o oră până a se decărca complet. Rata



Figura III.26: Servomotorul HD1900A.

de descărcare a acumulatorului este de 15C, ceea ce înseamnă că poate furniza un curent maxim de 19,5 de Amperi, mai mult decât suficient pentru mașina prototip. Greutatea acumulatorului este de 400 de grame, fiind suficient de ușor pentru a fi montat pe vehicul fără a suprasolicita motorul. Alegerea acestui tip de acumulator a fost motivată de caracteristicile acestuia: greutate redusă, curent mare de ieșire și rată de descărcare scăzută. Singurul dezavantaj al acumulatorilor de tip LiPo îl prezintă defectarea la descărcarea excesivă. Dacă tensiunea scade sub 3 Volți pe celulă, acumulatorul se poate distruge. Trebuie implementată o funcție de protecție, măsurând în permanență tensiunea de pe acumulator și având grijă să nu scadă sub 6 Volți în cazul acumulatorilor cu 2 celule în serie. Funcția de protecție poate fi implementată folosind convertorul analogic-digital de pe microcontroler, nefiind necesară măsurarea tensiunii la o frecvență mare.

A doua placă este opțională și poate fi folosită atunci când algoritmi de control implementați pentru plutonul de vehicule necesită o putere de calcul mai mare. Unul dintre scopurile principale ale proiectului a constat în folosirea metodei MPC pentru a controla distanța față de vehiculul aflat în față. A doua placă a fost aleasă ținând cont de rezultatele obținute de către alți cercetători ce au implementat controlul prin metoda MPC. Metoda MPC necesită putere mare de calcul, astfel încât microcontrolere pe 32 de biți cu arhitectură ARM (Advanced RISC Machine) reprezintă un punct de plecare. În [259] este descrisă abordarea metodei MPC bazată pe un microcontroller ARM cu Cortex-M3 ce rulează la 72 MHz. Metoda MPC descrisă în [84] implică un microcontroler ARM 9, obținând o perioadă maximă de eșantionare de 800 Hz (1.25 ms). La [38], folosind un microcontroler bazat pe o arhitectură ARM7 s-a obținut o perioadă de eșantionare de 250 Hz (4ms). Un mod complet automat de a implementa un controller MPC pe un sistem încorporat plecând de la un model dinamic în MATLAB cu o perioadă de eșantionare de ordinul kHz-ilor este prezentată la [85].

În urma acestor considerații, placa secundară ce formează sistemul încorporat al mașinii prototip este placa STM32F3DISCOVERY (Fig. III.27). Aceasta folosește microcontrolerul STM32F303VCT6, având 256KB de memorie flash și 48KB de memorie RAM. Microcontrolerul STM32F303VCT6 este bazat pe arhitectura Cortex-M4, având magistrala de date pe 32 de biți și instrucțiuni de tip C.I.S.C (Complex Instruction Set Computers). Are o frecvență de 72 MHz și o unitate pentru operații în virgulă mobilă. Avantajul faptului că are integrată această unitate constă în rapiditatea cu care se efectuează operațiile cu numere reale. Timpul de execuție al instrucțiunilor scade considerabil, împreună cu dimensiunea codului. Microcontrolerul suportă sistem de operare de timp real. Placa are încorporat depanatorul ST-LINK/V2, pentru depanarea aplicației încorporate prin USB. Kitul se alimentează de la USB, atunci când este conectat la PC sau extern, cu 3V sau 5 V, fapt ce constituie un mare avantaj deoarece poate fi alimentat direct din regulatoarele de tensiune de pe placa principală. Placa mai conține două butoane, unul pentru RESET și unul ce poate fi programat de utilizator și 7 led-uri ce sunt folosite pentru a indica starea aplicației și taskurile ce rulează la un anumit moment de timp. Mediul de dezvoltare folosit pentru programarea plăcii este IAR Embedded Workbench.

III.2.1.3 Proiectarea și realizarea plăcii principale cu microcontroller

Pentru placa principală a sistemului încorporat au fost luate în calcul două microcontrolere de la firma Microchip: dsPIC33FJ128MC708 și dsPIC33FJ128MC804. Principala

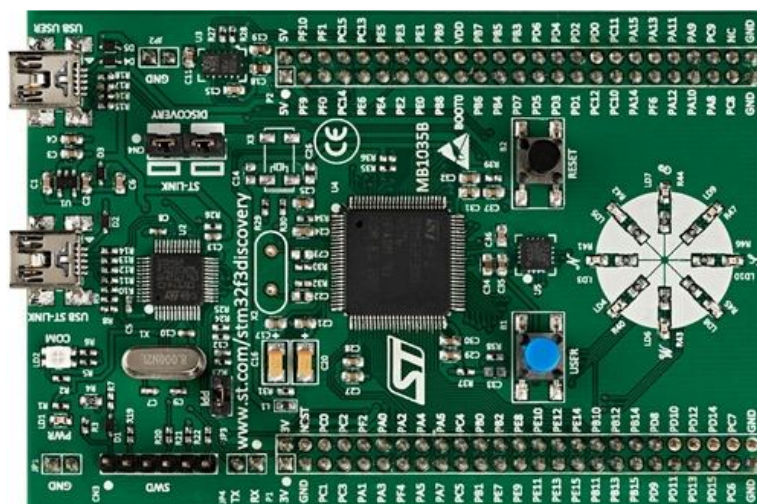


Figura III.27: Placa secundară, de tip STM32F3DISCOVERY.

diferență dintre ele constă în numărul de pini disponibili: dsPIC33FJ128MC708 are 80 de pini pe capsulă, din care 69 pot fi folosiți ca Intrări/Ieșiri, iar dsPIC33FJ128MC804 are 44 de pini pe capsulă din care 35 pot fi folosiți ca Intrări/Ieșiri, majoritatea fiind și remapabili. A fost ales dsPIC33FJ128MC804, fiind suficiente numărul de pini și perifericele oferite. Având două interfețe SPI și una I2C este suficient pentru SD Card, accelerometru și modulul de comunicație radio sau placa secundară. Pentru măsurarea vitezei de rotație a motorului folosind encoderul, dsPIC-ul are încorporat un periferic pentru encodere cu quadratură. Semnalul PWM necesar pentru servomotor și pentru placa de forță ce va comanda motorul DC poate fi obținut folosind două din cele 4 generatoare de PWM ale microcontrolerului, cu o rezoluție de 25ns. Pentru a măsura tensiunea bateriei și distanța de la senzorul de distanță POLOLU-2476 dsPIC-ul are un convertor ADC ce poate fi configurat pentru o rezoluție de 10 sau 12 biți și până la 9 intrări analogice. Majoritatea pinilor sunt reconfigurabili (26) ușurând astfel aranjarea componentelor la realizarea cablajului. De asemenea, dsPIC-ul are încorporat și un modul pentru comunicația pe CAN, având nevoie doar de un transceiver. Sunt disponibile și 8 canale DMA(Direct Memory Access) ce pot fi folosite pentru UART, SPI, ADC, CAN, INT0. Pentru a economisi spațiu pe placă tipul de capsulă ales a fost varianta SMD (QFN Quad Flat No-leads).

Primul pas după alegerea senzorilor și a microcontrolerului a fost alocarea unui anumit număr de pini pentru fiecare componentă (Fig. III.28). S-a ținut cont de perifericul necesar consultând foaia de catalog a microcontrolerului cât și de posibilitatea folosirii aceluși port pentru a conecta un alt senzor. În Fig. III.29 se poate vedea pinul corespunzător fiecărui senzor sau modul din componența sistemului încorporat.

Schema electrică a fost realizată în Eagle. Inițial pentru alimentare au fost folosite două regulatoare de tensiune. Primul regulator este un L7805, ce poate primi o tensiune maximă de 35V la intrare, iar la ieșire furnizează 5V, cu un curentul maxim ce poate fi generat de 1.5 A. Al doilea regulator este destinat alimentării microcontrolerului și anumitor senzori ce necesită 3.3V, regulatorul de tensiune folosit fiind un LM3940. Acesta la o tensiune de

| Componentă | Pini necesari microcontroler |
|-------------------------|------------------------------|
| Motor DC | 0 – conectat la driver |
| Driver motor | 1 DIG OUT, 1 PWM, 1 INT(opt) |
| Encoder motor | 2 DIG IN |
| Servomotor | 1 PWM |
| Senzor linie | 6 pini DIG IN/OUT, 1 DIG IN |
| Senzor obstacol | 1 DIG IN |
| Senzor distanță | 1 AN IN, 1 DIG OUT |
| Modul NRF24L01 | 4 SPI + 1 DIG OUT |
| Busolă și accelerometru | 2 I2C, 1 INT(opt) |
| Adaptor MicroSD | 4 SPI + 1 DIG IN |
| Tensiune Baterie | 1 AN IN |
| LED | 1 DIG OUT |

Figura III.28: Contorizare pini microcontroller.

intrare între 4.5 V și 5.5 V generează la ieșire 3.3V la un curent maxim generat de 1 A, suficient pentru microcontroler și un alt eventual consumator (Fig. III.30).

Deoarece motorul de curent continuu destinat propulsiei poate consuma până la 1.6 A (atunci când este blocat) și servomotorul 830mA, a fost necesară introducerea unui alt regulator pentru servomotor și pentru motorul de curent continuu. Regulatorul de tensiune ales este L7806. Acesta are la ieșire o tensiune de 6V iar curentul maxim generat este de 1.5 A. Alimentarea plăcii de forță pentru motorul DC se poate face fie de la regulatorul de 6V, fie direct de la baterie, acest lucru fiind setat prin intermediul jumper-ului J2 (Fig. III.31). Au fost prevăzute de asemenea și condensatoare de filtrare la intrarea și la ieșirea regulatorului, conform foii de catalog, pentru a avea o valoare a tensiunii de ieșire stabilă. Pentru a fi ușor de observat starea alimentării au fost adăugate două LED-uri de control, pentru linia de alimentare cu 3.3 Volți și respectiv pentru cea de alimentare cu 6 Volți. Pentru 5 Volți nu a fost necesar deoarece regulatorul de tensiune de 3.3 Volți este alimentat de la cel de 5 Volți iar dacă cel de 3.3 Volți funcționează, funcționează și cel de 5 Volți.

Alimentarea microcontrolerului a fost proiectată conform foii de catalog și a fost adăugat și un conector ICSP (In Circuit Serial Programming) destinat programării microcontroller-ului. Au fost prevăzuți condensatori de decuplare pentru fiecare pereche de pini Vdd/Vss și AVdd/AVss. Recomandarea producătorului este ca aceștia să aibă valoarea de 100 nF, fiind recomandați condensatori ceramici. De asemenea, aceștia au fost așezați cât mai aproape de pini microcontrolerului și au fost plasați pe aceeași parte cu microcontrolerul. Lungimea maximă a treseului dintre pin și condensator este de 6mm. Pin-ul Vcap nu trebuie conectat la Vdd și trebuie să aibă un condensator între 4.7 uF și 10 uF conectat la masă. Acesta poate fi ceramic sau tantalum. Alimentarea convertorului ADC prin pini AVdd și AVss trebuie să fie separată de tensiunea ce se dorește a fi măsurată, dar pentru această aplicație este necesară măsurarea tensiunii bateriei. Foia de catalog a microcontroller-ului recomandă folosirea unei bobine între pini AVdd și VDD pentru a îmbunătăți filtrarea

| Pin | Functions | | | | | | | Assignment |
|-----|-----------|---------|---------|---------|-------|-------|-------|-----------------------------|
| | Fct.1 | Fct.2 | Fct.3 | Fct.4 | Fct.5 | Fct.6 | Fct.7 | |
| 1 | SDA1 | RP9(1) | CN21 | PMD3 | RB9 | - | - | I2C Accelerometru |
| 2 | PWM2H1 | RP22(1) | CN18 | PMA1 | RC6 | - | - | S. dist.Enable IR/IRQ NRF |
| 3 | PWM2L1 | RP23(1) | CN17 | PMA0 | RC7 | - | - | Aux/INT1 Acc.: JP3 |
| 4 | RP24(1) | CN20 | PMA5 | RC8 | - | - | - | Aux/Fault Encoder : JP4 |
| 5 | RP25(1) | CN19 | PMA6 | RC9 | - | - | - | Encoder motor |
| 6 | VSS | - | - | - | - | - | - | |
| 7 | VCAP | - | - | - | - | - | - | |
| 8 | PGED2 | PWM1H3 | RP10(1) | CN16 | PMD2 | RB10 | - | Programming pin |
| 9 | PGEC2 | PWM1L3 | RP11(1) | CN15 | PMD1 | RB11 | - | Programming pin |
| 10 | PWM1H2 | DAC1RP | RP12(1) | CN14 | PMD0 | RB12 | - | Driver motor PWM |
| 11 | PWM1L2 | DAC1RN | RP13(1) | CN13 | PMRD | RB13 | - | SPI Data IN for card reader |
| 12 | TMS | PMA10 | RA10 | - | - | - | - | LED |
| 13 | TCK | PMA7 | RA7 | - | - | - | - | NRF Chip Enable DIG OUT |
| 14 | PWM1H1 | DAC1LP | RTCC | RP14(1) | CN12 | PMWR | RB14 | Servomotor PWM control |
| 15 | PWM1L1 | DAC1LN | RP15(1) | CN11 | PMCS1 | RB15 | - | Encoder motor |
| 16 | AVDD | - | - | - | - | - | - | |
| 17 | AVSS | - | - | - | - | - | - | |
| 18 | MCLR | - | - | - | - | - | - | |
| 19 | AN1 | VREF+ | CN2 | RA0 | - | - | - | Tensiune baterie |
| 20 | AN0 | VREF- | CN3 | RA1 | - | - | - | S. dist. Infr. intrare AN |
| 21 | PGED1 | AN2 | C2IN- | RP0(1) | CN4 | RB0 | - | SPI Data OUT for NRF |
| 22 | PGEC1 | AN3 | C2IN+ | RP1(1) | CN5 | RB1 | - | SPI Clk for NRF |
| 23 | AN4 | C1IN- | RP2(1) | CN6 | RB2 | - | - | SPI CS for NRF |
| 24 | AN5 | C1IN+ | RP3(1) | CN7 | RB3 | - | - | SPI Data IN for NRF |
| 25 | | DAC1RM | RP16(1) | CN8 | RC0 | - | - | Senzor linie |
| 26 | | DAC1LM | RP17(1) | CN9 | RC1 | - | - | Senzor linie |
| 27 | AN8 | CVREF | RP18(1) | PMA2 | CN10 | RC2 | - | Senzor linie |
| 28 | VDD | - | - | - | - | - | - | |
| 29 | VSS | - | - | - | - | - | - | |
| 30 | OSC1 | CLKI | CN30 | RA2 | - | - | - | Aux 1/External CLK |
| 31 | | CLKO | CN29 | RA3 | - | - | - | Aux 1/External CLK |
| 32 | | PMA8 | RA8 | - | - | - | - | Senzor linie LED |
| 33 | SOSCI | RP4(1) | CN1 | RB4 | - | - | - | SD Card SPI CS |
| 34 | SOSCO | T1CK | CN0 | RA4 | - | - | - | SD Card Detect DIG IN |
| 35 | TDI | PMA9 | RA9 | - | - | - | - | Driver motor dir |
| 36 | RP19(1) | CN28 | PMBE | RC3 | - | - | - | Senzor linie |
| 37 | RP20(1) | CN25 | PMA4 | RC4 | - | - | - | Senzor linie |
| 38 | RP21(1) | CN26 | PMA3 | RC5 | - | - | - | Senzor linie |
| 39 | VSS | - | - | - | - | - | - | |
| 40 | VDD | - | - | - | - | - | - | |
| 41 | PGED3 | ASDA1 | RP5(1) | CN27 | PMD7 | RB5 | - | SD Card SPI Clk |
| 42 | PGEC3 | ASCL1 | RP6(1) | CN24 | PMD6 | RB6 | - | SD Card SPI Data OUT |
| 43 | INT0 | RP7(1) | CN23 | PMD5 | RB7 | - | - | Senzor obstacol infrarosu |
| 44 | SCL1 | RP8(1) | CN22 | PMD4 | RB8 | - | - | I2C Accelerometru |

Figura III.29: Corespondența dintre pinii microcontrolerului și pinii senzorilor și modulelor.

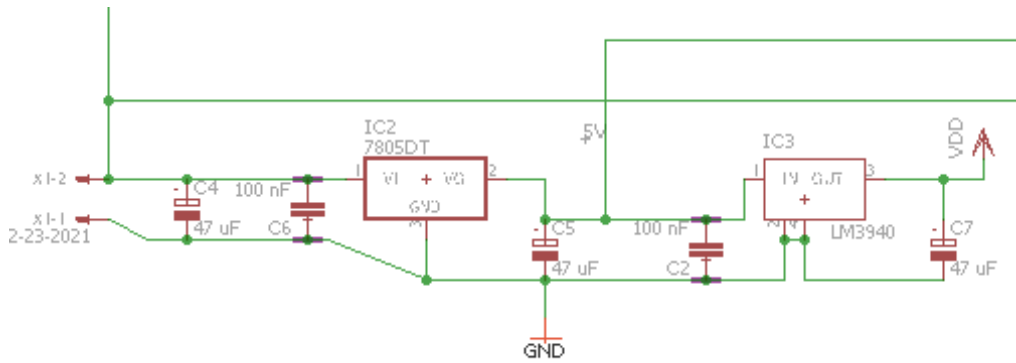


Figura III.30: Varianta de alimentare cu reglatoare de 5V și 33V.

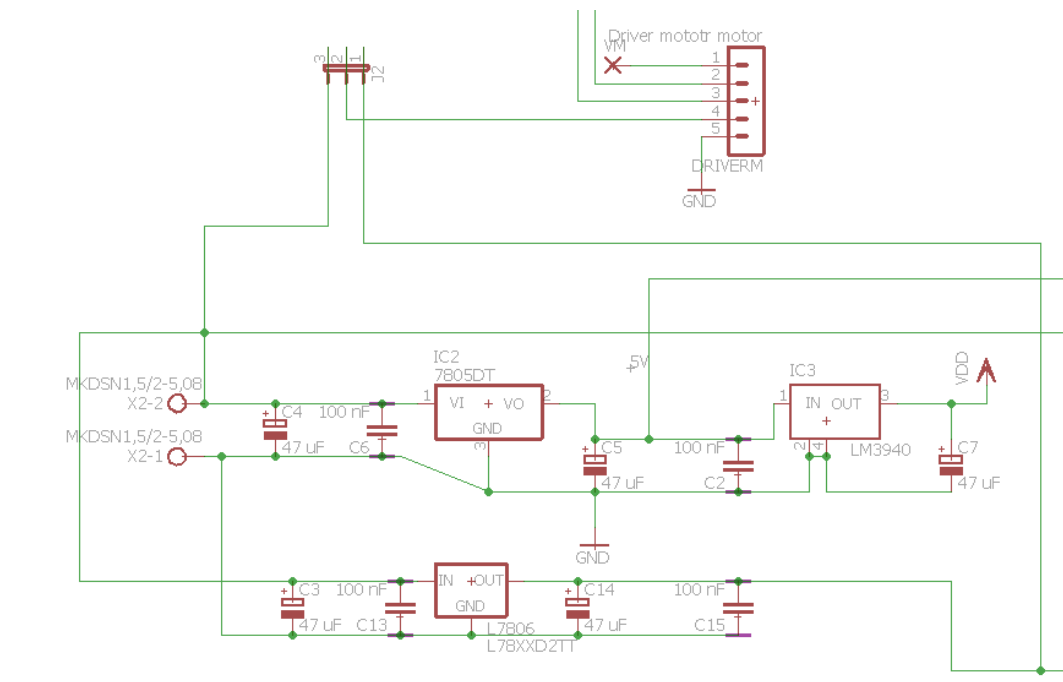


Figura III.31: Varianta de alimentare cu reglatoare de 6V, 5V și 3.3V.

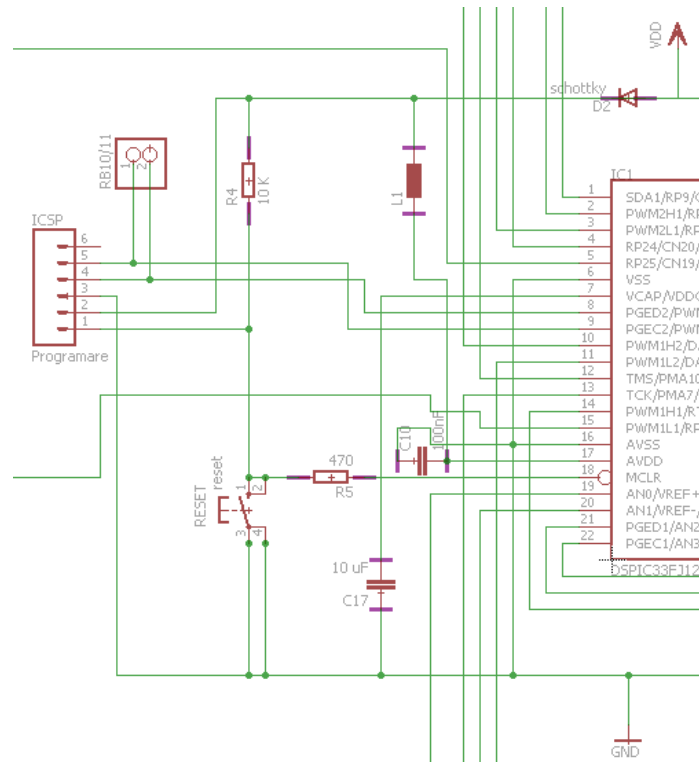


Figura III.32: Circuitele de programare și de reset ale microcontroller-ului.

convertorului ADC. Impedanța acesteia trebuie să fie sub 1 . Pin-ul MCLR are două funcții specifice: pentru resetarea microcontrolerului și pentru programare și debug. În timpul programării microcontrolerului rezistența și capacitatea ce pot fi adăugate pinului trebuie să fie cât mai mici. Programatorul/debugger-ul modifică starea pinului efectuând tranziții rapide de semnal ce nu trebuie afectate. Este recomandată izolarea pinului printr-un jumper față de restul circuitului în timpul programării/depanării. Pentru pinii ICSP sunt trei opțiuni pe placă, configurabili din biții de configurare. A fost aleasă perechea de pini PGED2 și PGEC2 (pinii 8 și 9) ținând cont și de lungimea traseului până la conectorul ICSP (se recomandă să fie cât mai scurt). Pentru a putea fi totuși folosiți și în alte scopuri, s-a ales conectarea acestora la un jumper, astfel încât atunci când nu se programează sau nu se face depanare pe microcontroler, aceștia să poată fi folosiți ca fiind pini reconfigurabili. Conectorul ICSP precum și circuitul de reset al microcontroller-ului sunt prezentați în Fig. III.32.

Pentru a măsura nivelul de tensiune al acumulatorului și a implementa funcția de protecție la descărcare necesară pentru acumulatorii de tip LiPo a fost folosit un divizor rezistiv. Prin intermediul pinului AN0, aparținând portului A al microcontrolerului este preluată tensiunea de pe una dintre rezistențele divizorului. Acesta este necesar deoarece tensiunea maximă permisă pe pinii microcontrolerului este de 3,3 Volți. Valorile rezistențelor sunt de 10 Kiloohmi și 30Kiloohmi, având un raport de 1/4 și au fost alese astfel încât să poată fi folosite și cu acumulatori LiPo cu două celule sau cu trei celule conectate în serie. Pe pinul RA10 al portului A a fost adăugat un LED ce poate fi folosit pentru a indica starea

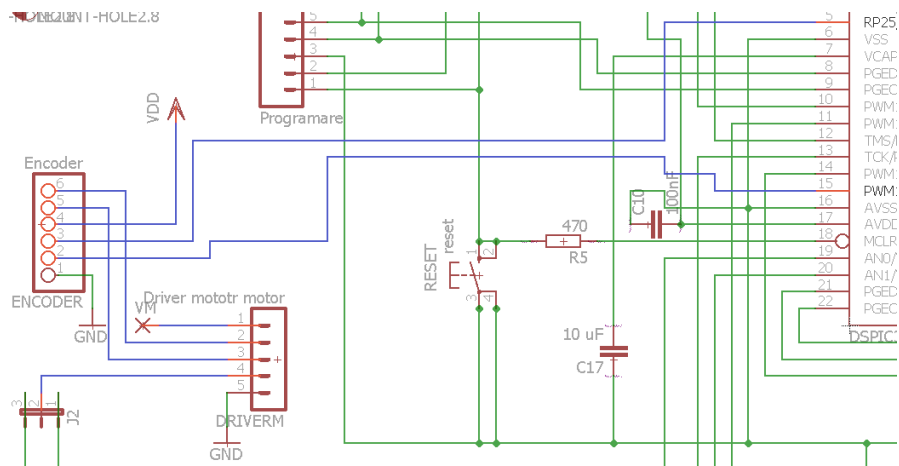


Figura III.33: Circuitul de conectare al plăcii de forță la motor.

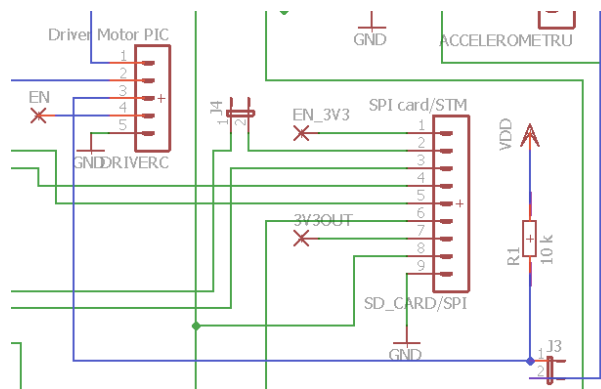


Figura III.34: Circuitul de conectare al plăcii de forță la microcontroller.

aplicației sau pentru depanare. Dacă LED-ul nu este necesar, pinul poate fi folosit și în alte scopuri, acest lucru fiind permis de jumper-uli J6.

Placa de forță cu MAX14870 este conectată pe placa principală prin intermediul a două baghete cu 5 pini. Encoder-ul magnetic este atașat motorului, toate conexiunile acestuia cu placa principală fiind realizate printr-un cablu panglică. Pe o parte a plăcii de forță se află distribuții pinii pentru alimentare și ieșirile către motor iar pe cealaltă parte intrările și ieșirile către microcontroller. Pe partea cu microcontrollerul PWM-ul de la microcontroller pentru controlul vitezei motorului este scos pe pinul 10 (PWM1H2), iar pin-ul pentru controlul direcției motorului este RA9. Encoder-ul mai are și un pin de întrerupere în caz de eroare. Pin-ul ENABLE de la driver nu necesită a fi controlat, el este lăsat în 0 logic, placa de forță fiind mereu activă. Driver-ul mai are un pin, FAULT ce va fi conectat la masă (low) în caz de supracurent sau în caz de încălzire excesivă. Circuitele de conectare a plăcii de forță cu motorul și microcontrollerul sunt prezentate Fig. III.33 și respectiv Fig. III.34.

Servomotorul este alimentat de la regulatorul de tensiune de 6V iar controlul acestuia se realizează prin intermediul pinului RB14 configurat ca ieșire PWM. Servomotorul se conectează direct la placă prin intermediul conectorului său cu 3 pini, prezentat în Fig.

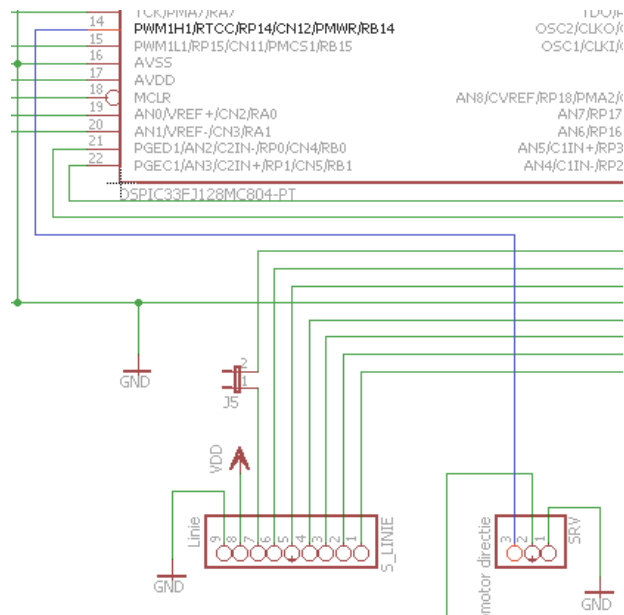


Figura III.35: Circuitul de conectare al servomotorului la microcontroller.

III.35.

Senzorul de linie este alimentat de regulatorul de 3.3 V. Pentru ca mașina prototip să poată urmări linia sunt suficiente 6 celule din cele 8 disponibile ale senzorului, senzorului de linie i-au fost asociați 6 pini conectați la port-ul C (RC0-RC5) ce pot fi configurați ca intrări sau ieșiri digitale. Senzorul mai are disponibil un pin prin intermediul căruia se poate opri LED-ul atunci când senzorul nu este folosit pentru a reduce consumul de curent sau se poate modifica intensitatea LED-ului prin PWM, dar folosirea acestui pin este opțională. Pe placă este conectat la pinul RA8 prin intermediul unui jumper (J5). Dacă nu se dorește oprirea alimentării LED-urilor pinul poate fi folosit în alte scopuri (Fig. III.36).

Senzorul de distanță POLULU 2476 este conectat la microcontroller prin intermediul pinului analogic AN0. Pinul EN al senzorului este opțional și poate fi folosit pentru a opri emițătorul cu infraroșu, trecând astfel senzorul în modul de consum redus. El este conectat la VCC prin intermediul unei rezistențe de pull-up de 10 k. Pentru că este opțional și va fi folosit rar, nefiind necesar modul de consum redus, pinul EN este conectat prin intermediul unui jumper cu trei pini (J1) la pinul RC6 al microcontrollerului. Pe cealaltă poziție a jumper-ului se poate selecta ca la pinul RC6 al microcontrollerului să fie conectată întreruperea de la NRF, pinul RC6 fiind și un pin reconfigurabil (Fig. III.37).

Modulul de comunicație nRF24L01 comunică cu microcontrollerul prin interfața SPI, primul fiind conectat la pini reconfigurabili ai microcontroller-ului. MOSI este conectat la RP3, CS este conectată la RP2, SCK este conectat la RP1 și MISO este conectat la RP0. Pin-ul CE (Chip Enable) este conectat la ieșirea digitală RA7. Pin-ul IRQ este conectat la RC6 (RP22) prin intermediul jumper-ului J1. Cu acest jumper se poate schimba intrarea la pinul microcontrollerului între întreruperea de la nRF și pinul ENABLE de la senzorul de distanță. Sunt posibile trei surse de întreruperi pentru pinul IRQ, configurabile din registrul CONFIG al modulului nRF.

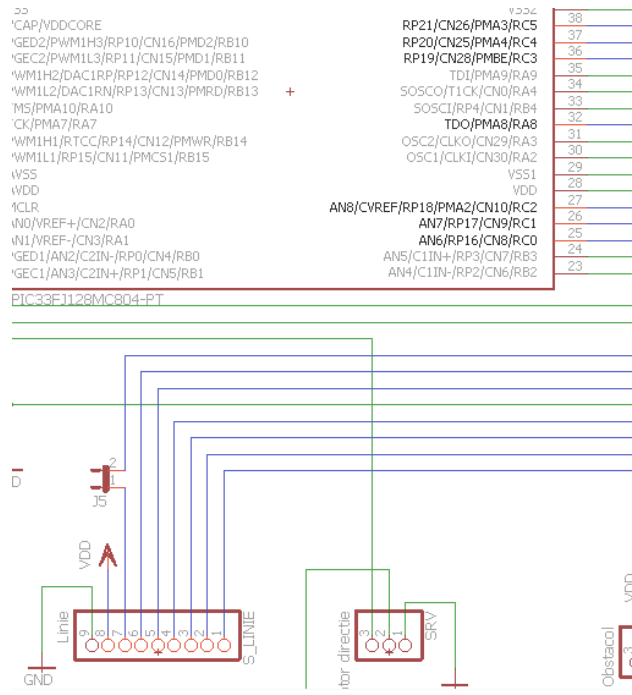


Figura III.36: Conectarea senzorului de linie la microcontroller.

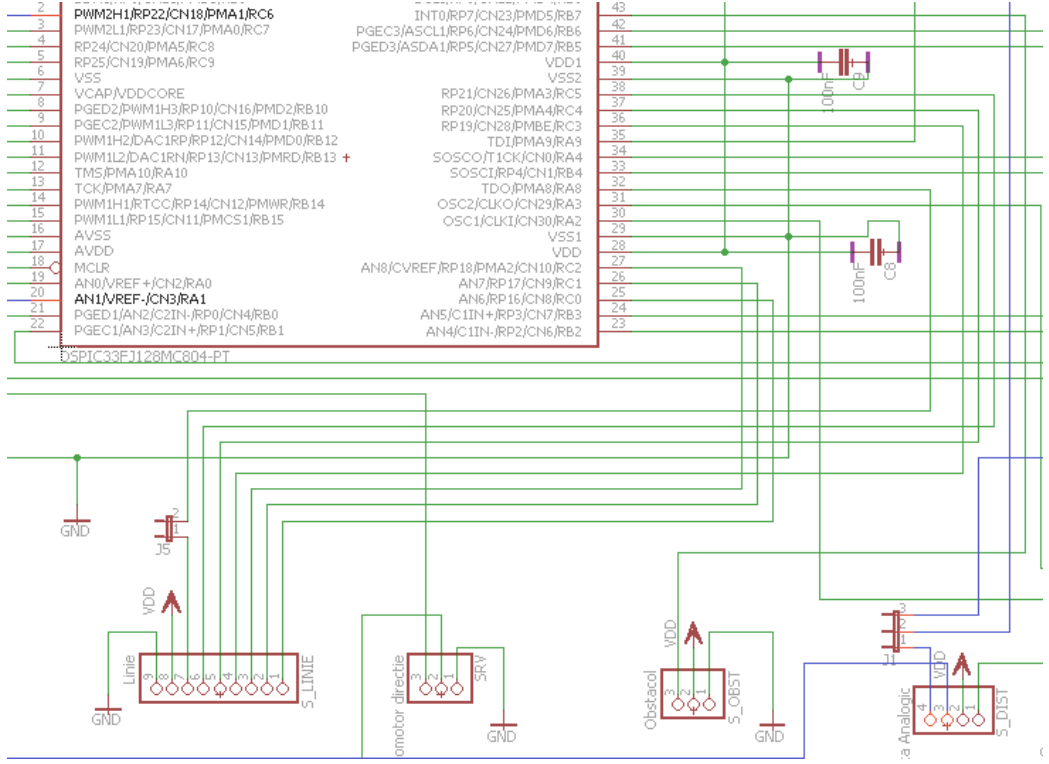


Figura III.37: Circuitul de conectare a senzorului de distanță la microcontroller.

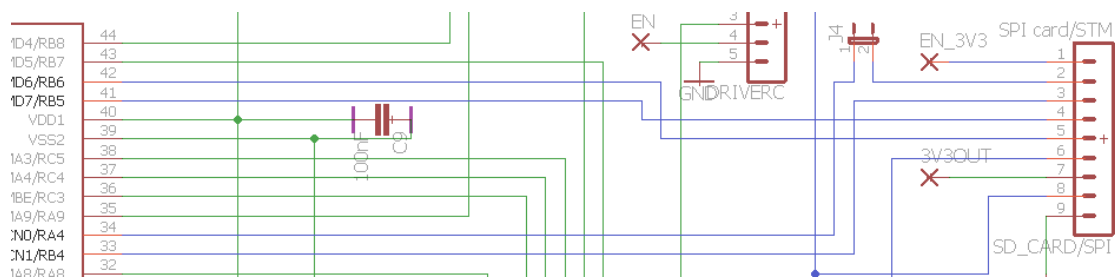


Figura III.38: Circuitul de conectare al adaptorului pentru microSD card la microcontroller.

Adaptorul pentru SD card este conectat pe placa principală prin intermediul unei baghete cu 9 pini (Fig. III.38). Acesta comunică tot prin interfața SPI cu microcontrolerul. Adaptorul are un pin ce semnalizează introducerea cardului în slot prin intermediul unui microcontact din interiorul slot-ului pentru card. Atunci când cardul este introdus pinul va fi conectat la VDD, iar când nu este prezent nici un card este conectat la masă. Utilizarea lui este opțională deoarece prezența cardului poate fi detectată folosind driver-ul ce încearcă să comunice cu cardul. Dacă cardul nu răspunde la comenzi înseamnă că nu este prezent sau este defect. Din acest motiv pinul este conectat la jumper-ul J4, făcând posibilă utilizarea pinului RA4 de la microcontroller și în alte scopuri. Pini EN și 3V3 nu sunt conectați la microcontroller deoarece nu sunt folosiți, dar pot fi conectați oricând prin intermediul unor fire deoarece conexiunile au fost realizate pe placă. Pin-ul EN pornște regulatorul de 3.3 V de pe adaptor iar ieșirea acestuia poate fi folosită prin intermediul pinului 3V3. Adaptorul este alimentat direct de la 3.3V deci nu este nevoie de regulatorul intern al acestuia.

Accelerometrul și busola comunică cu microcontrolerul prin interfața I2C, astfel fiind necesari doar doi pini. La microcontroler sunt folosiți pini SDA1 (I2C Data Line) și SCL1 (I2C Clock Line), pini pentru interfața I2C nefiind reconfigurabili. La această magistrală I2C mai pot fi conectate și alte periferice folosind conectorul I2C (Fig. III.39). Prin intermediul jumperilor SDA și SCL accelerometrul și busola pot fi deconectate de la magistrala I2C dacă utilizarea acestora nu este necesară. Este de asemenea posibilă conectarea accelerometrului și a busolei la interfața SPI, toți pini necesari având conectori pe placă. Selectarea magistralei dintre SPI și I2C se face prin intermediul pinului CS (Chip Select). Dacă acesta este lăsat în starea lui inițială (conectat la VDD) senzorul va comunica prin I2C, dacă CS este conectat la masă senzorul va recepționa comenzi pe interfața SPI. Mai sunt disponibili doi pini pentru întreruperi date de senzori dar nefiind ambii necesari unul a fost conectat la microcontroller (RP23) iar celălalt la un pin extern, putând fi conectat ulterior. Pe acești pini pot fi configurate întreruperi pentru detectarea căderii libere, detectare câmp magnetic sau detecție de mișcare.

Microcontrollerul are un oscilator intern cu o frecvență de 7.37 MHz, frecvența de lucru a microcontroller-ului putând fi adusă până la 80MHz prin intermediul modulului PLL. În schema plăcii principale este prevăzut și un oscilator extern împreună cu cei doi condensatori de 22 pF necesari. Dacă nu se folosește oscilatorul extern, cei doi pini pot fi folosiți ca intrări sau ieșiri digitale (nu sunt pini reconfigurabili) fiind conectați la pini 5 și 6 ai mufei AUX1 (Fig. III.40).

La realizarea cablajului au fost folosite două straturi, cablajul fiind dublu-placat. Straturile

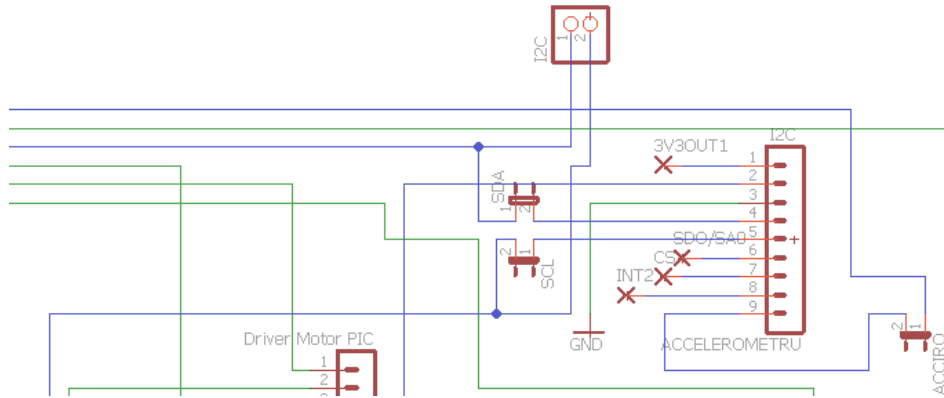


Figura III.39: Circuitul de conectare al modului LSM303D la magistrala I2C.

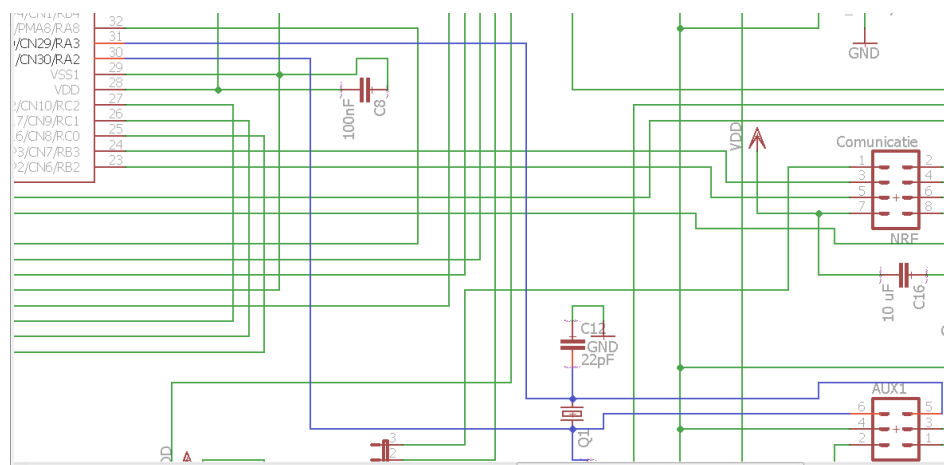


Figura III.40: Circuitul de conectare a oscilatorului extern precum și a pinilor auxiliari.

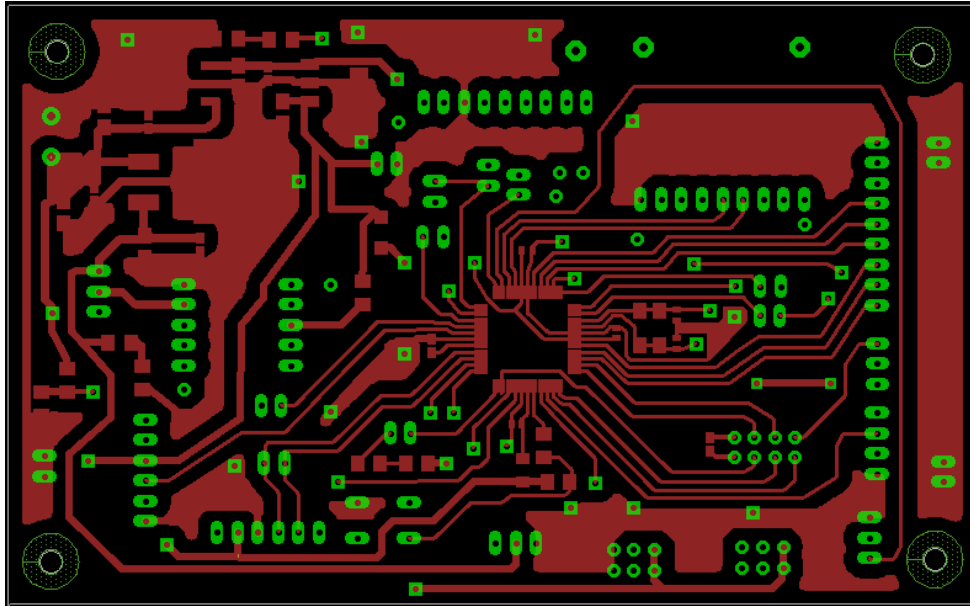


Figura III.41: Stratul "Top" cablajului imprimat.

sunt numite în Eagle "Top" (Fig. III.41) și "Bottom" (Fig. III.42).

Toate componentele, inclusiv microcontrolerul, sunt plasate pe stratul "Top", ce reprezintă partea de sus a plăcii. Traseele au o grosime de 0.8 mm pentru partea de alimentare a servomotorului și a driver-ului pentru motor, iar pentru traseele de semnal ale microcontrolerului grosimea traseului este de 0.4 mm pentru a se potrivi și cu dimensiunea pad-urilor microcontrolerului. Pentru trasee nu au fost folosite unghiuri de 90 de grade evitând astfel apariția problemelor atunci când pe acel traseu sunt semnale cu frecvență mare. Pe celălalt strat (partea de sub microcontroler, "Bottom") o mare suprafață este ocupată de planul de masă (sub microcontroler și sub reglatoarele de tensiune).

Primul prototip de cablaj pentru testarea funcționalităților a fost realizat prin metoda expunerii la lumină UV (Fig. III.43). Versiunea finală a cablajului cu senzorii montați este prezentată în Fig. III.44.

III.2.1.4 Prezentarea aplicației încorporate destinate identificării

Pentru a putea identifica modelul matematic pentru mașina prototip, a fost realizată o aplicație încorporată dedicată extragerii informațiilor de viteză. Nu a fost folosit sistemul de operare de timp real FreeRTOS deoarece s-a dorit ca sistemul încorporat să nu fie încărcat, pentru a extrage și stoca informațiile de viteză cu o frecvență cât mai mare. Impulsurile primite de la encoderul magnetic generează cereri pentru întreruperea externă INT1, producând în final incrementări ale contorului de impulsuri. Valoarea acestui contor de impulsuri contribuie la determinarea distanței parcurse de mașină într-un interval dat de timp. Inițializarea și tratarea întreruperii INT1 este prezentată în 1.

Listing 1: Inițializarea și tratarea întreruperii INT1

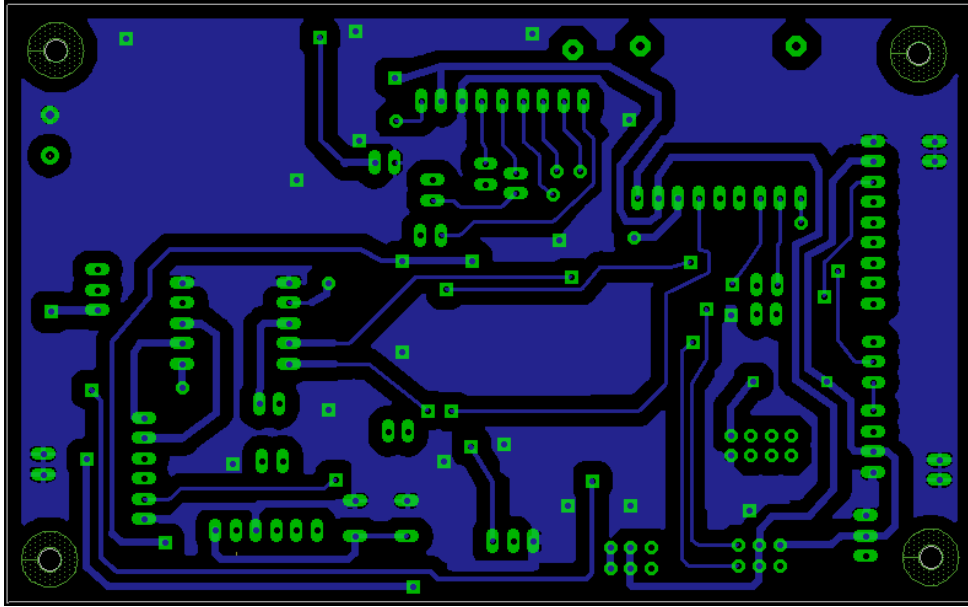


Figura III.42: Stratul "Bottom" cablajului imprimat.

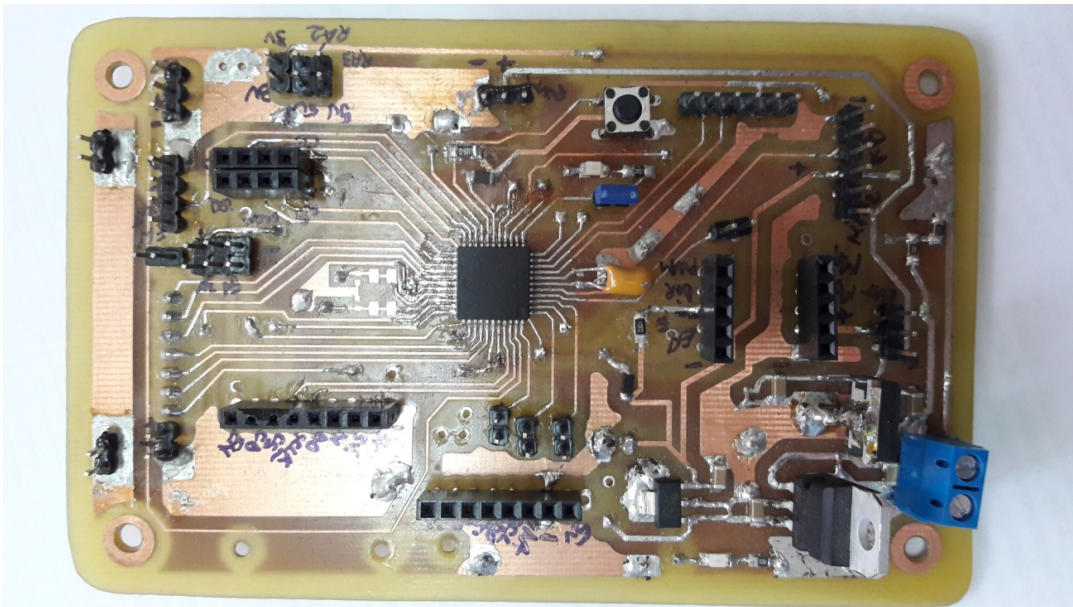


Figura III.43: Prototipul cablajului imprimat.

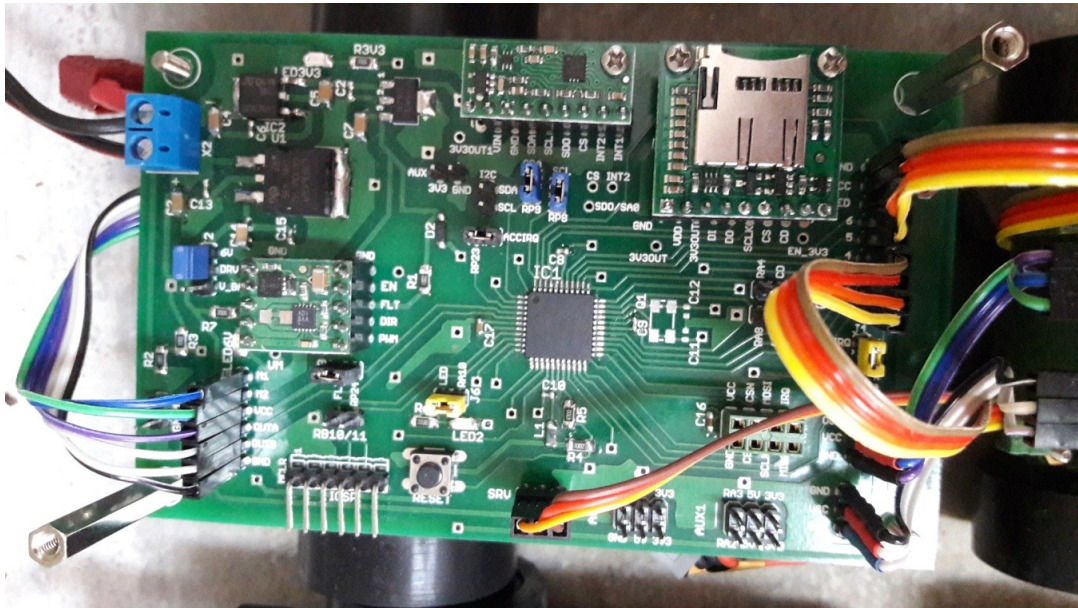


Figura III.44: Versiunea finală a plăcii principale.

```

void prvSetup_INT1_Interrupt (void)
{
    RPINR0bits.INT1R = 25; // OUTA - RP25

    _INT1EP = 0;           // intrerupere generata pe front pozitiv
    _INT1IF = 0;          // achita intreruperea
    _INT1IE = 1;          // permite lucrul cu intreruperea INT1
}

void __attribute__((interrupt, no_auto_psv)) _INT1Interrupt ( void )
{
    enc_countA++;         // incrementeaza contorul de impulsuri
    _INT1IF = 0;         // achita intreruperea
}

```

Pentru a determina viteza curentă a mașinii prototip, este necesară determinarea distanței parcurse de mașină într-un interval de timp cunoscut și fixat. Intervalul de timp este măsurat cu ajutorul canalului de timp T2, ce generează o întrerupere la fiecare 10ms. Rutina de tratare a întreruperii canalului de timp T2 se ocupă cu salvarea valorii contorului de impulsuri într-o variabilă auxiliară și resetarea valorii contorului de impulsuri. Inițializarea și tratarea întreruperii canalului de timp T2 este prezentată în 2.

Listing 2: Inițializarea și tratarea întreruperii T2

```

// timer T2 - genereaza intrerupere la fiecare 10ms
static void prvSetup_Timer_T2_Interrupt ( void )
{
    T2CON = 0;
    TMR2 = 0;
}

```

```

PR2 = ( unsigned short ) 6250; // 10 ms cu prescaler de 1:64

_T2IF = 0; // achita intreruperea

_T2IE = 1; // permite lucrul cu intreruperea de la timer T2

T2CONbits.TCKPS = 2; // prescaler 1:64

T2CONbits.TON = 1; // porneste timer
}

int enc_count, enc_countA, flag;
void __attribute__((interrupt, no_auto_psv)) _T2Interrupt( void )
{
    enc_count = enc_countA; // salveaza numarul de impulsuri

    enc_countA = 0; // reseteaza contorul de impulsuri

    flag = 1; // pt. identificare

    _T2IF = 0; // achita intrerupere
}

```

Vectorul cmd contine informații privind momentele de timp la care se schimbă comanda aplicată motorului de tracțiune precum și valoarea comenzii aplicate. Informațiile privind valoarea vitezei mașinii la fiecare 10ms sunt salvate în vectorul viteze în prima parte a codului din funcția main, parte prezentată în 3.

Listing 3: Salvarea informațiilor de viteză în memorie

```

// 10ms
unsigned int cmd[32][2]={{0,0},
    {100,25000},{150,50000},{200,65000},{250,40000},{300,30000},
    {350,35000},{400,25000},{450,65000},{500,40000},{550,0},
    {600,25000},{650,50000},{700,65000},{750,40000},{800,30000},
    {850,35000},{900,25000},{950,65000},{1000,40000},{1050,0},
    {1100,25000},{1150,50000},{1200,65000},{1250,40000},{1300,30000},
    {1350,35000},{1400,25000},{1450,65000},{1500,40000},{1550,0},
    {1600,0}
};

unsigned int crt_cmd;
time = -1; crt = 0;

while(time<1560)
{
    if (flag==1) // au trecut 10ms
    {
        i = enc_count*40; // 10ms

        time++; // incrementeaza contorul de
                perioade

        viteze[time] = i;

        if (time==cmd[crt][0])
        {
            // transmite comanda la motor
            P1DC2 = cmd[crt][1];
            // salveaza valoarea curenta a comenzii
            crt_cmd = cmd[crt][1];
            // pregateste indicele pentru urmatoarea comanda

```

```

        crt++;
    }

    comenzi[time] = crt_cmd/10;

    flag = 0;
}
}

```

Partea a doua a funcției main se ocupă cu stocarea pe cardul SD a informațiilor privind valorile corespunzătoare comenzii și vitezei slavate în memorie în etapa anterioară și este prezentată în 4.

Listing 4: Salvarea informațiilor din memorie pe SD card

```

rc = f_open(&Fil, "file.txt", FA_WRITE | FA_OPEN_APPEND);
for(i=0;i<1560;i++)
{
    if (viteze[i]!=0)          // viteza diferita de zero
    {
        _itoaQ15(viteze[i],sir);
        // salveaza viteza
        rc = f_write(&Fil, (const void*)sir, 3, &bw );
    }
    else                       // viteza nula
        // salveaza viteza
        rc = f_write(&Fil, (const void*)"0", 1, &bw );
    rc = f_write(&Fil, (const void*)" ", 1, &bw );

    if (comenzi[i]!=0)        // comanda diferita de zero
    {
        _itoaQ15(comenzi[i],sir);
        // salveaza comanda
        rc = f_write(&Fil, (const void*)sir, 4, &bw );
        rc = f_write(&Fil, (const void*)"0 ", 2, &bw );
    }
    else                       // comanda nula
    {
        rc = f_write(&Fil, (const void*)"0 ", 2, &bw );
    }
}
rc = f_close(&Fil);

```

A fost realizat un program în Matlab pentru a putea încărca și vizualiza datele salvate pe SD card de aplicația încorporată prezentată anterior.

Pentru fiecare mașină din pluton a fost realizată achiziția datelor necesare identificării. În Fig. III.45 este ilustrată comanda aplicată motorului de tracțiune, în Fig. III.46 este prezentat cum se modifică viteza mașinii prototip, iar Fig. III.47 prezintă viteza filtrată a mașinii prototip.

Măsurătorile achiziționate experimental au fost folosite în toolbox-ul System Identification (Ident) în mediul Matlab pentru a obține un model al vehiculelor. Modelele au fost obținute ca funcții de tranfer de forma:

$$G(s) = \frac{K}{T_s + 1}, \quad (\text{III.1})$$

unde K este factorul de amplificare și T este constanta de timp a sistemului.

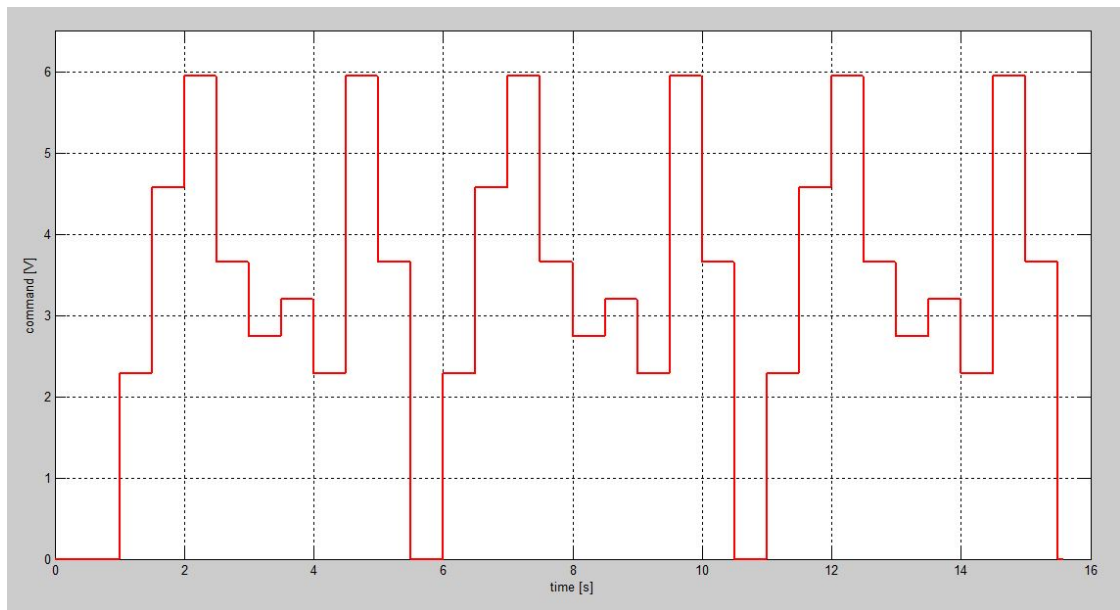


Figura III.45: Valoarea comenzii aplicate motorului de tracțiune.

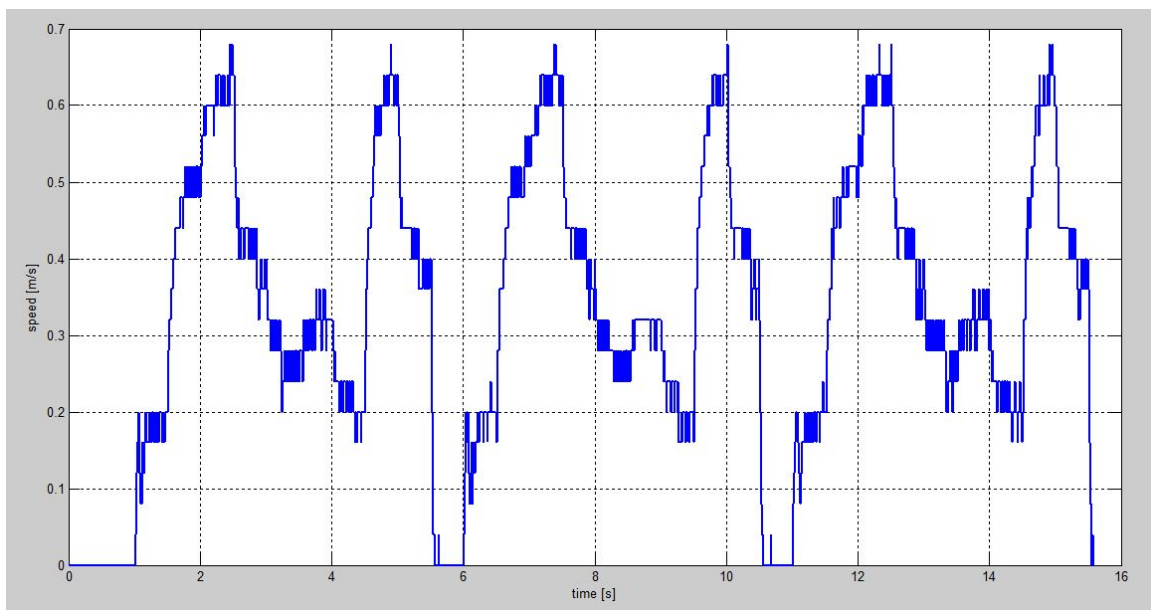


Figura III.46: Valoarea vitezei mașinii prototip.

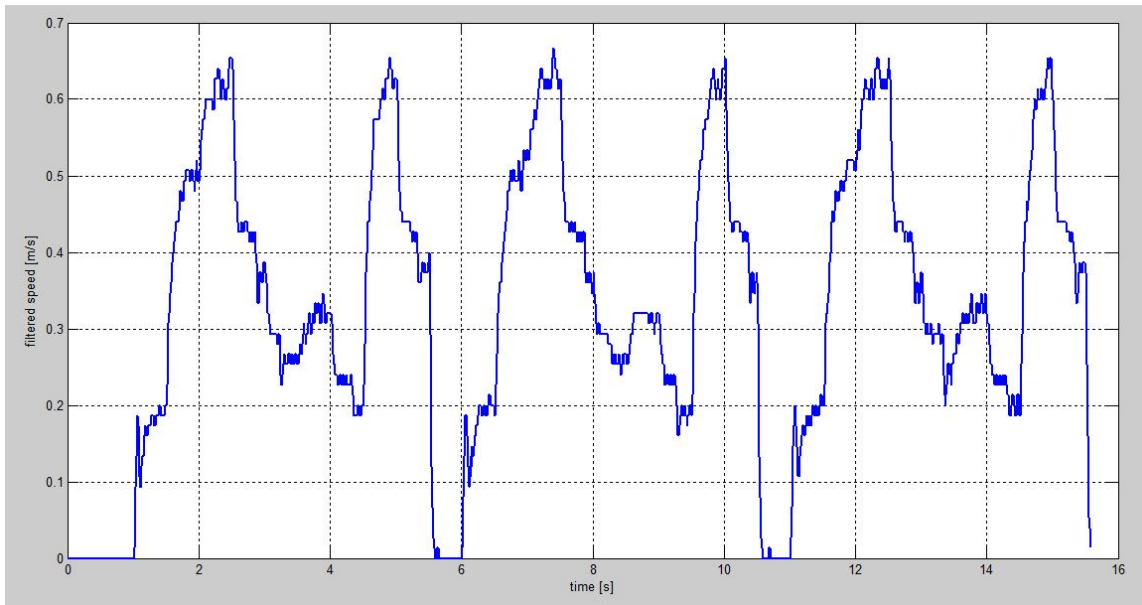


Figura III.47: Valoarea vitezei filtrate a mașinii prototip.

În figura III.48 se pot vedea răspunsurile la semnale de tip treaptă ale celor 8 modele obținute pentru fiecare vehicul în parte și, de asemenea, răspunsul unui model obținut ca medie între toate modelele (evidențiat cu roșu) pentru care $K = 0.1088$ și $T = 0.0752$.

III.2.1.5 Prezentarea aplicației încorporate bazate pe FreeRTOS

Au fost efectuate mai multe experimente pentru a demonstra principalele funcționalități ale soluției propuse. Aplicația încorporată bazată pe FreeRTOS a fost direcționată către un microcontroler dsPIC33FJ128MC802 produs de Microchip. Microcontrolerul este compus din periferice standard pe chip, incluzând șase interfețe de comunicații seriale asincrone (SCI) și un modul ECAN compatibil cu CAN 2.0 A, B. . Microcontrolerul dsPIC este capabil să se ocupe de toate task-urile prezentate anterior dacă cele mai rapide nu au cerințe pentru o putere de procesare ridicată. Dat fiind că unul dintre obiectivele proiectului este implementarea controlului longitudinal cu un controler MPC, utilizarea plăcii STM32F3DISCOVERY este absolut necesară. Această placă are un microcontroler de 32 biți cu memorie flash de 256kB, memorie RAM de 48kB și poate rula la frecvența maximă de 72MHz. Placa STM32F3DISCOVERY va comunica cu placa principală prin interfața serială (SPI). Mai mult decât atât, deoarece controlerul longitudinal utilizează informațiile de la senzorul cu raze infraroșii, senzorul poate fi conectat direct la a doua placă. Modulul de comunicație RF sau cardul microSD ar putea fi conectați la placa secundară, de asemenea, dacă este necesar. Modelul prototip al plutonului de vehicule echipat cu placa de bază este prezentat în Figura 7.

Aplicația încorporată a plăcii principale a fost concepută astfel încât să respecte cerințele de sincronizare ale senzorilor și modulelor folosite. Senzorul de linie, un QTR-8RC este capabil să furnizeze informații cu o frecvență de 500 Hz. Accelerometrul LSM303D poate

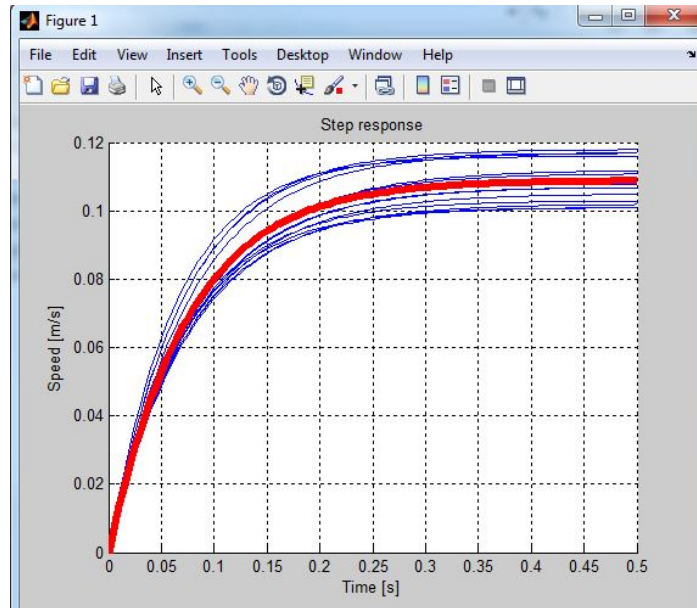


Figura III.48: Experiment identificare.

fi interogat prin magistrala I2C la 100 kHz în modul standard și la 400 kHz în modul rapid. Senzorul de proximitate infraroșu este un Sharp GP2Y0D810Z0F care are o rată tipică de eșantionare de aproape 400 Hz. Perioada maximă de actualizare a senzorului de distanță analogic Sharp GP2Y0A60SZLF este de 20 ms.

Aplicația încorporată poate fi croită și modificată în funcție de cerințele experimentelor. Funcțiile de inițializare a modului ADC precum și a canalului de timp T3 sunt prezentate în 5. Canalul de timp T3 este setat să starteze conversii AD cu perioada de 25ms. Dat fiind ca achiziția se realizează succesiv pe doua canale AD (unul dedicat senzorului de distanță și unul dedicat tensiunii acumulatorului), achiziția AD de pe canalul corespunzător senzorului de distanță se realizează la 50ms.

Listing 5: Funcții inițializare modul ADC și canal de timp T3

```
// functie initializare modul ADC
void init_Adcl(void)
{
    AD1CON1bits.AD12B = 1; // conversie AD pe 12 biti
    AD1CON1bits.FORM = 0; // rezultat conversie integer
    AD1CON1bits.SSRC = 2; // timerul 3 starteaza conversia
    AD1CON1bits.ASAM = 1; // incepe esantionarea noii valori imediat dupa
                          // terminarea
                          // unei conversii

    //AD1CON2bits.CSCNA = 1; // scaneaza intrarile pe CH0+ in timpul
    // achizitiei A
    AD1CON2bits.CHPS = 0; // converteste doar CH0
    AD1CON2bits.SMPI = 1; // incrementeaza adresa DMA dupa terminarea
                          // fiecarei
                          // conversii 1-conv.alt. 0 -
                          // conv. singulara

    AD1CON2bits.ALTS = 1; // alternate A si B
}
```

```

AD1CHS0bits.CHOSA=0; // MUXA +ve input selection (AIN0) for CH0
AD1CHS0bits.CHONA=0; // MUXA -ve input selection (Vref-) for CH0
AD1CHS0bits.CHOSB=1; // MUXB +ve input selection (AIN1) for CH0
AD1CHS0bits.CHONB=0; // MUXB -ve input selection (Vref-) for CH0

AD1CON3bits.ADRC = 0; // foloseste ceasul magistralei
AD1CON3bits.ADCS = 63; // Timpul necesar unei conversii este de 19.2 us
                        // Ceasul pentru conversia AD
                        // are formula Tad=Tcy*(adcs
                        // +1)
                        // Tad=Tcy*(adcs+1)=(1/40)
                        // *64=1.6us

// Se seteaza intrarile analogice AN1
//AD1CSSLbits.CSS1 = 1; // Selectam intrarea analogica AN1(RA1) pentru a
// fi scanata

// Scriem registrul de configurare al portului
// Se va folosi doar registrul low al portului de configurare deoarece dsPIC33fj128MC802
// nu are implementati mai mult de 6 pini pentru ADC
AD1PCFGL = 0xFFFF; // Setam toti pinii portului ADC1 pe modul digital
                    // si activeaza citirea la
                    // intrarea portului

_TRISA1 = 1; // tensiune senzor distanta
_TRISA0 = 1; // tensiune baterie
AD1PCFGLbits.PCFG0 = 0; // Setam pinul AN0(RA0) pe intrare analogica,
                        // ADC verifica voltajele
                        // pe acel pin (achizitie
                        // AD)
AD1PCFGLbits.PCFG1 = 0; // Setam pinul AN1(RA1) pe intrare analogica,
                        // ADC verifica voltajele
                        // pe acel pin (achizitie
                        // AD)
IFS0bits.AD1IF = 0; // Reseteaza flag-ul intreruperii convertorului AD
IPC3bits.AD1IP = 6; // Seteaza prioritatea intreruperii convertorului
                    // AD
IEC0bits.AD1IE = 1; // Permite intreruperea convertorului AD

AD1CON1bits.ADON = 1;
}

// Timer-ul 3 este setat sa starteze conversia AD la fiecare 25 milisecunde.
void init_Tmr3()
{
    TMR3 = 0;
    PR3 = 3906;
    T3CONbits.TCKPS = 3; // 1:256
    T3CONbits.TON = 1; // Start Timer 3
}

```

În 6 sunt prezentate tabela de tip "look-up" folosită pentru determinarea distanței corespunzătoare unei tensiuni măsurate la ieșirea senzorului de distanță precum și rutina de tratare a întreruperii ADC.

Listing 6: Rutina de tratare a întreruperii ADC și tabela look-up pentru distanta

```

int ain4Buff[SAMP_BUFF_SIZE];
int sampleCounter=0;
unsigned int canal = 0;
float tens, dist;

```

```

// Look-up table pentru distanta
int vLT[2][21] =
    { // cod digital
      {2424, 2257, 2065, 1957, 1840, 1760, 1637, 1578, 1490, 1435,
        1382, 1309, 1272, 1235, 1205, 1173, 1130, 1107, 1083, 1062, 1036},
      // milimetri
      {100, 110, 120, 130, 140, 150, 160, 170, 180, 190, 200,
        210, 220, 230, 240, 250, 260, 270, 280, 290, 300}
    };

// rutina de tratare a intreruperii convertorului AD
void __attribute__((interrupt, no_auto_psv)) _ADC1Interrupt(void)
{
    int i;
    int reset_ref=0;
    if (canal==0) // 0 - baterie
    {
        tens = ADC1BUF0 * 3.24 / 4096;
        if (tens<1.4)
        {
            P1DC2 = 0; // opreste tractiunea
            P1DC1 = 7500; // centreaza directia
        }
    }
    else // 1 - senzor distanta
    {
        for (i=0; i<20; i++)
        {
            if ((ADC1BUF0<=vLT[0][i]) && (ADC1BUF0>vLT[0][i+1])) // gasesc
                intervalul
                { // determina distanta
                    dist = vLT[1][i+1]-((vLT[1][i+1]-vLT[1][i]) * ((ADC1BUF0-vLT
                        [0][i+1]) * 1.0 / (vLT[0][i]-vLT[0][i+1])));
                    reset_ref=1;
                }
        }
        if(reset_ref==0)
        {
            if(ADC1BUF0<1036)
                dist=310; //
            else if(ADC1BUF0>2424)
            {
                dist=90;
            }
        }
    }

    canal^=1;

    IFS0bits.AD1IF = 0; // Achita intreruperea convertorului AD
}

```

Task-ul Line_S_Task este prezentat în 7 și se ocupă cu detectarea poziției liniei ce indică traseul ce trebuie urmărit de către vehicul.

Listing 7: Task-ul Line_S.Task ce detectează poziția liniei față de senzor

```

int val;

// senzor linie - detecteaza linia
void Line_S_Task(void *params)
{
    int SL_AND_MASK, SL_OR_MASK, SL_NR_R_BITS;
}

```

```

SL_AND_MASK = 0b1111111111000000;    // 0 pe poz pinilor
SL_OR_MASK = 0b0000000000111111;    // 1 pe poz pinilor
SL_NR_R_BITS = 0;
for (;;)
{
    // facut pinii RC5-RC0 iesiri
    TRISC = TRISC & SL_AND_MASK;
    // pus 1 pe pini
    PORTC = PORTC | SL_OR_MASK;
    DelayUs(20);    // delay c.p. 10us

    // facut pinii RC5-RC0 intrari
    TRISC = TRISC | SL_OR_MASK;
    vTaskDelay(1);    // delay 1ms

    // citit valori
    val = (PORTC & SL_OR_MASK) >> SL_NR_R_BITS;
    vTaskDelay(9);
    // perioada totala 10ms
}
}

```

Regulatorul PID implementat pentru urmărirea liniei este implementat în task-ul LineFoll_Task și este prezentat în 8.

Listing 8: Task-ul LineFoll_Task

```

int cp;
// line following controller
void LineFoll_Task(void *params)
{
    static int i, a[50], ref, err, perr, kP, kI, kD, P, I, D, cmd;

    for(i=0;i<50;i++) a[i] = 0;
    a[1] = 1; a[3] = 2; a[2] = 3; a[6] = 4; a[4] = 5;
    a[12] = 6;
    a[8] = 7; a[24] = 8; a[16] = 9; a[48] = 10; a[32] = 11;

    ref = 6; perr = 0;    // 6 corespunde liniei positionate
                        // pe mijlocul senzorului

    kP = 300;
    kI = 0.25;
    kD = 600;
    I=0;

    for (;;)
    {
        // PathPID
        if (a[val]>0 && a[val]<11)
        {
            cp = a[val];    // pozitia curenta
            err = ref - cp; // calculeaza eroarea
            P = err * kP;
            I = I + err;
            D = err - perr;

            cmd = 7500 + err*kP + D*kD + I*kI;

            // limiteaza comanda
            if(cmd >= 9000)

```

```

        cmd = 9000;
    if(cmd <= 6000)
        cmd = 6000;

    // aplica comanda servomotorului
    P1DC1 = cmd;
    // salveaza eroarea
    perr = err;
    }
    vTaskDelay(10);
    }
}

```

Codul sursă ce se ocupă de lucrul cu SD cardul este împărțit în 9 și în 10, primul conținând funcția de inițializare a modului microSD iar al doilea conținând task-ul ce se ocupă cu salvarea datelor pe SD card (MicroSD_Task).

Listing 9: Inițializarea modului microSD card

```

// functie initializare modul microSD
void Init_SDCard(void)
{
    // initializeaza interfata SPI
    Init_SPI2();
    DelayMs2(200);
    DelayMs2(200);
    // initializeaza SD cardul
    S=disk_initialize(0); // 1 - no ini
    if(S==STA_NOINIT) // 1 - no init
        {"Eroare la initializare"
        }
    else
        {"OK"
        }

    //rc=f_mount(0, &fatfs); //OLD
    rc=f_mount(&fatfs, "0", 1);
    rc=f_opendir(&dir, "\\");
    f_readdir(&dir, &fno);

    // creaza fisierul
    rc = f_open(&Fil, "file.txt", FA_WRITE | FA_OPEN_APPEND);
    rc = f_write(&Fil, (const void*)"RESET", strlen("RESET"), &bw );
    rc = f_close(&Fil);
}

```

Listing 10: Task-ul MicroSD_Task

```

extern int start_radio;
extern float dist;
extern long v_crt_mm, cmd_cV;

// taskul ce salveaza date pe SD card
void MicroSD_Task(void *params)
{
    int i;
    static char sir[4];

    for (;;)
    {
        if (start_radio == 1) // plutonul a primit comanda de START
        {

```

```

        if(ROLE==LEADER)           // masina are rol de leader?
        {
            //i = (int) v_crt_mm;
            i = (int) cmd_cV;
            //i = cp;
            _itoaQ15(i,sir);
            rc = f_open(&Fil, "file.txt", FA_WRITE | FA_OPEN_APPEND);

            // salveaza datele pe card
            rc = f_write(&Fil, (const void*)sir, 3, &bw );
            rc = f_write(&Fil, (const void*)" ", 1, &bw );

            rc = f_close(&Fil);
        }
        else // masina este follower
        {
            i = (int) dist;
            //i = cp;
            _itoaQ15(i,sir);
            rc = f_open(&Fil, "file.txt", FA_WRITE | FA_OPEN_APPEND);

            rc = f_write(&Fil, (const void*)sir, 3, &bw );
            rc = f_write(&Fil, (const void*)" ", 1, &bw );

            rc = f_close(&Fil);
        }
    }
    vTaskDelay(50);
}
}

```

În 11 este prezentată funcția de inițializare a modulului radio.

Listing 11: Inițializare modul radio

```

// initializare modul radio
void Init_Radio(void)
{
    Init_nRF24();           // setari predefinite nRF
    prvSetup_INT2_Interrupt(); // setare intrerupere nRF

    closeAllReadingPipes();

    if(CAR_ID>REPEATER_ID)  nRF24_Set_RF_CH(SEC_RF_CH);

    if(CAR_ID>=REPEATER_ID)
        openWritingPipe(CAR_ID-(REPEATER_ID-1));
    else
        openWritingPipe(CAR_ID);

    if(CAR_ID>=2 && CAR_ID<=REPEATER_ID)
        openReadingPipe(CAR_ID-1);
    else if(CAR_ID>REPEATER_ID)
        openReadingPipe(CAR_ID-REPEATER_ID);

    //if(ROLE==LEADER)
        openReadingPipe(0);

    startListening();      // trece in mod RX
    nRF24_IRQ_Clear_All();
}

```

Task-ul RF_TX_Task se ocupă cu transmiterea informațiilor prin radio și este prezentat în 12.

Listing 12: Task-ul RF_TX_Task

```

extern int start_radio;

//register to hold letter sent and received
unsigned char datatx[3]="ST";
//register to hold letter sent and received
unsigned char datarx[3]="bb";
volatile int data_sent=0;
unsigned char current_pipe=0;
// nRF24L01+
void RF_TX_Task(void *params)
{
    for (;;)
    {
        if(REPEATER)    nRF24_Set_RF_CH(SEC_RF_CH);    //schimba pe al
                    doilea canal radio
        stopListening();    //seteaza mod transmitator

        //transmite
        nRF24_Write_TX_Payload(datatx, 2, true);
        while(data_sent==0);    //asteapta terminarea transmisiei
        data_sent=0;    //reinitializeaza flag-ul

        startListening();    //seteaza mod receptor
        //schimba pe primul canal radio
        if(REPEATER)    nRF24_Set_RF_CH(nRF24_RF_CH_DEFAULT_VAL);

        if(ROLE==FOLLOWER)    // masina are rol de follower
            strcpy(datatx, "aa");

        if(ROLE==LEADER)    // masina are rol de leader
            if(strcmp(datatx,"ST")==0)
                _RA10=~_RA10;

        vTaskDelay(500);
    }
}

```

Recepția informațiilor primite prin radio este semnalizată de apariția întreruperii INT2. Rutina de tratare a întreruperii INT2 se ocupă cu recepția radio și este prezentată în 13.

Listing 13: Rutina de tratare a întreruperii INT2

```

//interrupt used for IRQ pin (nRF24)
void prvSetup_INT2_Interrupt(void)
{
    RPNR1bits.INT2R = 22;    //0x16; //assign external interrupt 2 to pin RP22
    nRF24_IRQ_DIR=1;    //Set IRQ pin as input
    _INT2EP=1;    //Interrupt on negative edge
    //_INT2IP=7; //Set priority from 0 to 7 (lowest to highest)
    _INT2IF=0;    //Reset flag
    _INT2IE=1;    //Enable external interrupt 2
}

//volatile unsigned int bckp_speed;
void __attribute__((interrupt, no_auto_psv)) _INT2Interrupt(void)
{
    if((nRF24_IRQ_Pin_Active() && nRF24_IRQ_RX_DR_Active()))    //receive
        interrupt
    {

```

```

//selecteaza pipe-ul pe care am primit
current_pipe=nRF24_Get_RX_Pipe();
nRF24_Read_RX_Payload(datarx, 2);

//pipe 0 - folosit doar de telecomanda pt comanda motorului
if(current_pipe==0)
{
    if( strcmp ( datarx,"ST" )==0 ) //start
    {
        start_radio=1;
    }
    else if(strcmp(datarx,"BK")==0 )//stop
    {
        P1DC2 = 0;
        start_radio = 0;
    }
}
else if( current_pipe!=0 && strcmp(datarx,"ST") ==0 )
{
    _RA10 = ~_RA10;
    //trimit pt urmatoarea masina
    strcpy(datarx,datarx);
}
nRF24_IRQ_Clear_RX_DR();
}
//transmit interrupt
else if((nRF24_IRQ_Pin_Active() && nRF24_IRQ_TX_DS_Active()))
{
    data_sent=1;
    nRF24_IRQ_Clear_TX_DS();
}
}
__INT2IF=0;
}

```

Dat fiind că același cod sursă este compilat și încărcat pe fiecare vehicul din pluton, diferențierea rolurilor de tip leader, follower și repeater se realizează prin intermediul setărilor din fișierul nRF_Config.h, prezentat în 14.

Listing 14: Fișierul de configurare a rolului mașinii în cadrul plutonului

```

#ifndef nRF_Config_H_
#define nRF_Config_H_

#define FOLLOWER      0
#define LEADER       1

#define CAR_ID        1 // 1-10
#define ROLE          LEADER
//#define ROLE        FOLLOWER

#define REPEATER      0 //0-no repeat 1-repeat
#define REPEATER_ID   6 //maxim 6

#define SEC_RF_CH     0x20 //canal radio pt coada plutonului (CAR_ID>=6)
#endif

```

Task-ul SpeedCtrl_PI_Task se ocupă cu implementarea regulatorului de viteză pentru leader și este disponibil în 15.

Listing 15: Task-ul SpeedCtrl_PI_Task


```

volatile int contor=0;
int start_radio=0;

extern int stop;
extern float dist;
extern int enc_count;
extern float v_ref;

// seteaza referinta pentru leader (viteza, in principiu)
void Start_Task(void *params)
{
    xTaskCreate(SpeedCtrl_PI_Task, (signed portCHAR *) "Ts9", 2*
        configMINIMAL_STACK_SIZE, NULL, tskIDLE_PRIORITY + 5, NULL);
    for (;;)
    {
        if (start_radio==1)
        {
            contor++;
            if (contor == 1)      P1DC2 = 25000;
            if (contor == 16)     P1DC2 = 30000;
            if (contor == 31)     P1DC2 = 35000;
            if (contor == 46)     P1DC2 = 30000;
            if (contor == 61)     P1DC2 = 25000;
        }
        vTaskDelay(1000);
    }
}

long v_crt_mm, cmd_cV;
float v_crt;
// regulator viteza pentru masina leader
void SpeedCtrl_PI_Task(void *params)
{
    static float err, perr, dif_sat, kP, kI, kWind, lP, lI;

    static float cmd, cmd_sat, pcmd;

    float Ts;

    // referinta de viteza pentru leader
    v_ref = 0;
    // initializari variabile auxiliare
    pcmd = 0;
    perr = 0;

    dif_sat = 0;

    // factorul de amplificare
    kP = 20;
    // factorul de integrare
    kI = 20;
    // factor anti-windup
    kWind = 20;
    // perioada de esantionare
    Ts = 0.05;

    for (;;)
    {
        if((stop == 0) && (start_radio == 1))
        {
            // det vit crt
            v_crt_mm = enc_count*8;          // encoder interogat la 50ms
            v_crt = v_crt_mm/1000.0;

            // eroarea de viteza pentru leader

```

```

// (v_lead_mas - viteza masurata a leaderului)
err = v_ref - v_crt;

// comanda P
lP = kP * err;

// comanda I (UI_1 - comanda I de la pasul anterior,
// e_l_1 - eroarea de viteza pentru leader de la pasul anterior)
lI = pcmd + Ts * kI * perr + Ts * kWind * dif_sat;

// comanda aplicata leaderului
cmd = lP + lI; // in volti
//cmd_cV = cmd*100;

if (cmd>6) cmd_sat = 6;
else if (cmd<0) cmd_sat = 0;
else cmd_sat = cmd;

cmd_cV = cmd_sat*100;

PlDC2 = cmd_sat*65535/6;

// diferenta saturatie
dif_sat = cmd_sat - cmd;
// salvare eroare anterioara
perr = err;

// salvare comanda anterioara
pcmd = lI;
}
else
{
}
vTaskDelay(50);
}
}

```

III.2.2 Implementarea algoritmilor de control în sistemul încorporat

III.2.2.1 Implementarea algoritmului de control PID

Task-ul LongCtrl_PID_Task implementează regulatorul PID de distanță, codul său sursă fiind disponibil în 16.

Listing 16: Task-ul LongCtrl_PID_Task

```

// task-ul de control al distantei cu PID
void LongCtrl_PID_Task(void *params)
{
    static long ref, cp, err, perr, kP, kI, kD, lP, lI, lD;

    static long cmd;

    ref = 150; // referinta ideala 110-130mm maxim 150mm
    perr = 0;

    kP = 150;
    kI = 7;
}

```

```

kD = 75;
lI=0;

for (;;)
{
    if((stop == 0) && (start_radio == 1))
    {
        // v2 - PID
        if (dist>=100 && dist<=300)
        {
            cp = dist;
            err = cp - ref;
            lP = err * kP;
            lI +=(kI * err);
            lD = err - perr;
            cmd = lP + lI + lD*kD;
            if(cmd>42000)
            {
                lI-=cmd-42000;
                cmd=42000;
            }
            else if(cmd<8000)
            {
                lI+=8000-cmd;
                cmd=8000;
            }
            P1DC2 = cmd;
            perr = err;
        }
        else // nu vede masina in fata
            if(P1DC2>=45000) // 55000 CROAZIERA
                P1DC2=P1DC2-5000; // SCAD LA
                CROAZIERA
            else if (P1DC2<44000) //55000+1000
                P1DC2=P1DC2+500; // cresc LA
                CROAZIERA
            else
                P1DC2=44500;
    }
    vTaskDelay(50);
}
}

```

III.2.2.2 Implementarea algoritmului de control CACC-GPC

Task-ul LongCtrl_GPC_Task implementează regulatorul GPC de distanță, codul său sursă fiind disponibil în 17.

Listing 17: Task-ul LongCtrl_GPC_Task

```

// date de intrare
// v_mas
// d_mas
void LongCtrl_GPC_Task(void *params)
{
    static float d_ref, d_ref_const, d_ref_var, dist_s, d_mas, d_mas_1, d_mas_2;
    static float v_mas, v_ant, v_ant_1, dif_v_ant, dif_v_ant_1;
    static float U_dir, U_dir_1, U_dir_2;
    static float hdi, e_f, U_perturb, U_perturb_1, U_perturb_2, U_aplicat_follow;
}

```

```

static float cmd_f, cmd_f_sat, cmd_f_cV;

U_perturb_1 = 0;
U_perturb_2 = 0;
U_dir_1 = 0;
U_dir_2 = 0;
v_ant = 0;
v_ant_1 = 0;
dif_v_ant_1 = 0;
d_mas_1 = 0;
d_mas_2 = 0;

// referinta de distanta pentru followeri constanta
d_ref_const = 0.15; // referinta ideala 110-130mm maxim 150mm

// distanta in stationare
dist_s = 0.11; // d min 10cm

// time headway
hdi = 0.1;

// referinta de distanta pentru followeri variabila cu viteza
d_ref_var = dist_s + v_mas * hdi;

d_ref = d_ref_const;
//d_ref = d_ref_var;

for (;;)
{
// nu e obstacol si s-a dat start din radio
if((stop == 0) && (start_radio == 1))
{
if (dist >= 100 && dist <= 300)
{
d_mas = dist/1000.0;
//v_ant = ;

////eroarea de distanta pentru followeri (
//d_mas - distanta masurata a followerului;
//d_mas_1 - distanta masurata a followerului la pasul
anterior;
//d_mas_2 - distanta masurata a followerului cu doi pasi
anteriori)
e_f = -737 * d_ref + 1843 * d_mas - 1475 * d_mas_1 + 369 *
d_mas_2;

//comanda directa (U_dir_2 - comanda directa cu doi pasi
anteriori)
U_dir = U_dir_2 + e_f;

//diferenta de viteza pentru vehiculul anterior (
//v_ant - viteza vehiculului anterior;
//v_ant_1 - viteza vehiculului anterior la pasul anterior)
dif_v_ant = v_ant - v_ant_1;

//comanda din perturbatii
//U_perturb_2 - comanda din perturbatii la 2 pasi
anteriori,
//dif_v_ant_1 - diferenta de viteza pentru vehiculul
anterior de la pasul anterior)
U_perturb = U_perturb_2 + 37 * dif_v_ant - 18 *
dif_v_ant_1;

//comanda aplicata followerului
cmd_f = U_dir + U_perturb;

```

```

        if (cmd_f>6) cmd_f_sat = 6;
        else if (cmd_f<0) cmd_f_sat = 0;
        else cmd_f_sat = cmd_f;

        cmd_f_cV = cmd_f_sat*100;

        P1DC2 = cmd_f_sat*65535/6;

        d_mas_2 = d_mas_1;
        d_mas_1 = d_mas;

        v_ant_1 = v_ant;

        dif_v_ant_1 = dif_v_ant;

        U_dir_2 = U_dir_1;
        U_dir_1 = U_dir;
        U_perturb_2 = U_perturb_1;
        U_perturb_1 = U_perturb;
    }
    vTaskDelay(50);
}
}

```

III.2.2.3 Implementarea algoritmului de control DMPC

Task-ul LongCtrl_DMPC_Task implementează regulatorul DMPC de distanță, codul său sursă fiind disponibil în 18.

Listing 18: Task-ul LongCtrl_DMPC_Task

```

// date de intrare
// v_mas
// d_mas
void LongCtrl_DMPC_Task(void *params)
{
    static float d_ref, d_mas, d_mas_1, d_mas_2;
    static float v_mas, v_ant, v_ant_1, dif_v_ant, dif_v_ant_1;
    static float e_f, U_aplicat_leader;
    static float cmd_f, cmd_f_sat, cmd_f_cV;

    v_ant = 0;
    v_ant_1 = 0;
    dif_v_ant_1 = 0;
    d_mas_1 = 0;
    d_mas_2 = 0;

    // initializari necesare algoritmului de control DMPC
    init_DMPC();

    // referinta de distanta pentru followeri constanta
    d_ref = 0.15; // referinta ideala 110-130mm maxim 150mm

    for (;;)
    {
        // nu e obstacol si s-a dat start din radio
        if((stop == 0) && (start_radio == 1))

```

```

    {
        if (dist>=100 && dist<=300)
        {
            d_mas = dist/1000.0;
            e_f = d_ref - d_mas;

            //calculeaza comanda aplicata followerului folosind DMPC
            cmd_f = calcul_DMPC(d_ref, d_mas, d_mas_1, d_mas_2, v_mas,
                v_ant, v_ant_1, dif_v_ant, dif_v_ant_1, e_f,
                U_aplicat_leader);

            if (cmd_f>6) cmd_f_sat = 6;
            else if (cmd_f<0) cmd_f_sat = 0;
            else cmd_f_sat = cmd_f;

            cmd_f_cV = cmd_f_sat*100;

            P1DC2 = cmd_f_sat*65535/6;

            d_mas_2 = d_mas_1;
            d_mas_1 = d_mas;

            v_ant_1 = v_ant;

            dif_v_ant_1 = dif_v_ant;
        }
    }
    vTaskDelay(50);
}
}

```

III.2.3 Algoritm DMPC în formulare cu model de viteză

În această secțiune se va prezenta implementarea și testarea algoritmului DMPC propus în secțiunea II.2.1, publicat în [30], aflat în revizie la [260], și reformulat în [146] pentru controlul unui pluton, folosind modelul vehiculelor de tipul element de ordinul întâi descris și identificat în secțiunea III.2.1.4.

Conform rezultatelor prezentate în secțiunea III.2.1.4, vehiculele implementate în cadrul acestui proiect pot fi modelate ca element de ordin întâi, cu constanta de timp $\tau = 0.1088$ și factorul de amplificare al sistemului $K = 0.0752$, plutonul putând fi modelat potrivit descrierii propuse în secțiunea II.2.2.2, atât pentru lider cât și pentru vehiculele urmăritoare. Algoritm de control DMPC, în care obiectivul de control a fost urmărirea unei referințe treaptă de viteză pentru lider, și distanță pentru urmăritori, a fost formulat potrivit descrierii din secțiunea II.2.1.3 bazat pe model de viteză.

În figura III.49 este afișată performanța de control obținută în simulare pentru un pluton alcătuit din $N = 8$ vehicule, într-un test de urmărirea referinței de viteză pentru lider, cu o schimbare de referință la momentul de timp $2s$, de la $0.3m/s$ la $0.5m/s$. Parametri de control folosiți sunt perioada de eșantionare $T_s = 0.01$ secunde, orizontul de predicție $N_p = 20$ eșantioane, orizontul de comandă $N_c = 5$ eșantioane, distanța impusă între vehicule de $0.01m$. Parametri de ponderare a comenzii sunt $R_{u_1} = 0.15$ pentru lider și $R_{u_i} = 0.00015$, $i = \overline{2, 8}$ pentru urmăritori. Pentru toate vehiculele, limitele restricțiilor pe intrare sunt stabilite conform specificațiilor tehnice ca $u_i^{\min} = 0$, $u_i^{\max} = 12$ V.

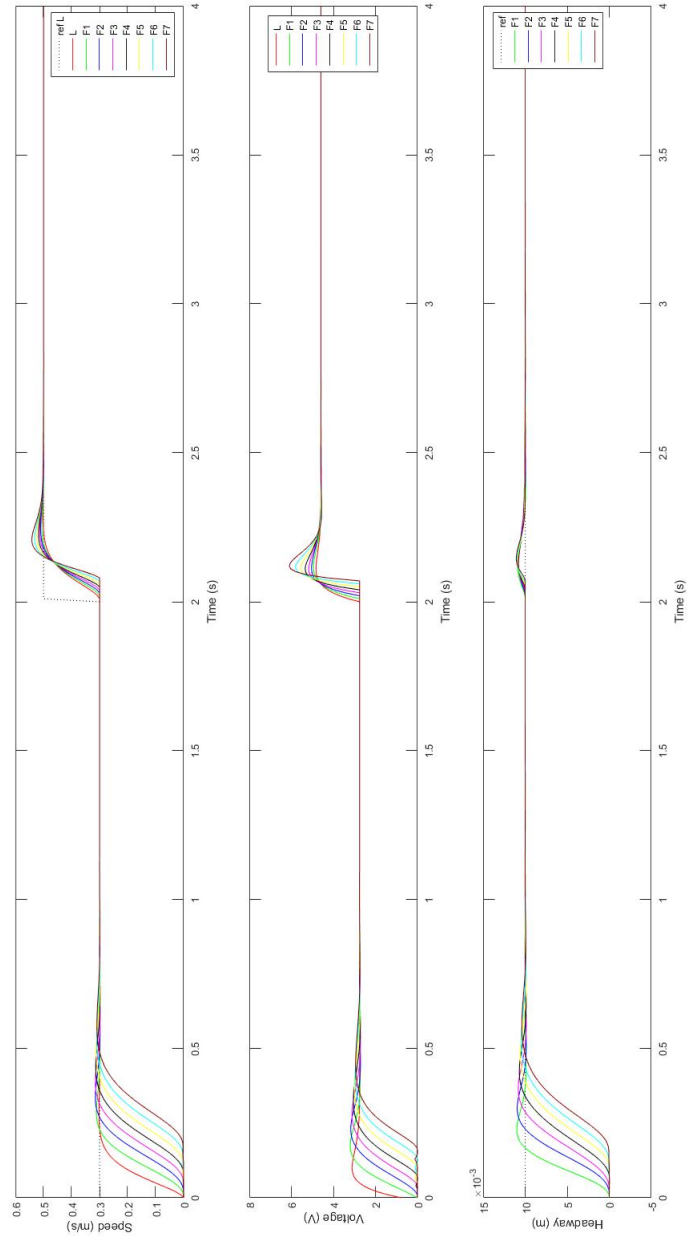


Figura III.49: Rezultate în simulare obținute pentru un pluton cu $N = 8$ vehicule modelate ca element de ordinul întâi.

III.3 (A3.2) Testarea strategiei de control predictiv distribuit pe un pluton de roboți mobili

III.3.1 Control predictiv descentralizat fără comunicații pentru o formație de roboți mobili

III.3.1.1 Introducere

Controlul unei formații de roboți mobili a atras atenția cercetătorilor de-a lungul anilor, fiind definit ca și coordonarea unui grup de roboți pentru a obține și menține o anumită structură a formației. Această problemă de control este de bază în coordonarea roboților mobili deoarece este inclusă în majoritatea cerințelor sistemelor multi-robot [261,262]. Coordonarea mișcării pentru sisteme multi-robot implică strategii pentru control poziției relative a fiecărui robot mobil raportată la ceilalți membri ai echipei [263]. În cadrul controlului unei formații, mișcarea roboților depinde de informația privitoare la poziție, deplasare sau distanța raportată la ceilalți roboți [264].

Cele mai recente abordări a controlului sistemelor multi-robot se concentrează pe tehnici distribuite sau descentralizate aplicate pe roboți în rețea [265]. În aceste sisteme multi-robot, fiecare robot are o rază limitată a senzorilor și a comunicației și pot folosi doar informații achiziționate de senzorii montați local sau recepționate de la vecinii direcți. În acest caz, principala problemă de control se referă la cum se pot deplasa roboții individuali folosind doar informații locale, satisfăcând cerințe globale raportate la configurația generală a sistemului [265]. În plus, proiectarea schemelor de control descentralizat pentru sisteme multi-robot are anumite provocări care nu se întâlnesc în controlul centralizat sau individual al roboților, e.g., interacțiuni complexe, paralelism implicit, dimensionalitatea mare a sistemului, informații incomplete, și incertitudini.

În literatura de specialitate disponibilă, sunt propuse două abordări pentru controlul unui pluton: fie considerând menținerea unei distanțe fixate între membrii plutonului, numită distanță independentă de timp [5, 13, 17, 19, 266–268] sau considerând că distanța dintre vehicule variază cu viteza membrilor plutonului, numită distanță dependentă de timp [22, 269, 270].

Rezultatele din [271] propun o schema de control descentralizat pentru roboți mobili de tip uniciclu care păstrează formația evitând coliziunile și obstacolele. În [17] se propune o strategie bazată pe regulatoare proporționale iar în [22] controlul este realizat cu tehnică de adaptare a parametrilor de acord ai regulatorului, aplicat pentru un pluton format din trei roboți mobili Pioneer 3AT. Rezultatele arată faptul că traiectoria ultimului robot din pluton poate diferi de traiectoria liderului, i.e., stabilitatea plutonului nu este asigurată. În [272] controlul longitudinal al unui pluton este validat prin intermediul unui regulator PI (cu o componentă de reacție), autorii incluzând și o componentă feed-forward, și anume viteza robotului predecesor, transmisă prin rețeaua de comunicație. Scheme lider-urmăritori pentru roboți modelați ca dublu integrator sunt studiate în [273]. În [261, 262] se analizează problema controlului descentralizat a unei formații de roboți mobili multi-agent bazată pe grafuri de formație. În [19, 267] sunt testate regulatoare în cascadă PI-P pentru controlul longitudinal și lateral aplicat pe roboți Surveyor SRV-1, dar proiectare regulatoarelor

nu a fost bazată pe metode de sinteză care să asigure stabilitatea plutonului. Rezultatele din [265] propun controlul descentralizat variant în time pentru o formație multi-robot cu dinamică de element de ordin întâi. În [13] sunt testate în simulare câteva regulatoare P și PI pentru roboți mobili mOway și rezultatele sunt comparate cu cele obținute cu un regulator bazat pe comportamentul șoferului. Pentru controlul descentralizat al unei formații lider-urmăritori, alcătuită din mulți roboți mobili autonomi este propusă o metodă de control sliding mode variabilă în timp [274]. Rezultatele din [275] propun un control optimal bazat pe orizontul alunecător, cu stabilitatea formației multi-robot garantată, cu evitarea coliziunilor și a obstacolelor, fiecare robot fiind descris cu dinamică de integrator.

În această secțiune se consideră ghidarea longitudinală descentralizată și controlul unui pluton de roboți mobili, i.e., în direcția de mișcare a plutonului, urmărind o linie marcată pe podea. Una dintre cele mai comune aplicații pentru un sistem lider-urmăritori, este controlul unui convoi alcătuit din roboți mobili autonomi. Soluția simplă oferită pentru problema controlului plutonului de roboți presupune că nu există comunicații între roboți și că fiecare robot are acces doar la distanța dintre el și robotul din față. Comunicația dintre roboți ar introduce o complexitate adițională, și o lipsă de fiabilitate în pluton. De-asemena ar complica proiectarea schemei de control și analiza stabilității. Eficacitatea strategiilor de control propuse s-a ilustrat utilizând un pluton de roboți mobili mOway, într-o configurație lider-urmăritori. Pentru fiecare robot mobil, s-au implementat două strategii predictive (i.e., control predictiv bazat pe stări - SS-MPC și control predictiv extins auto-adaptiv - EPSAC) și performanțele obținute în simulare pentru un pluton alcătuit dintr-un lider și 16 urmăritori au fost comparate cu cele obținute cu regulatoare clasice proporționale. În plus, câteva experimente au fost efectuate pentru a ilustra performanțele obținute în timp real, pe un pluton cu patru roboți mobili mOway (un lider și trei urmăritori). De-asemena, a fost analizată și stabilitatea plutonului, oferind-se informații utile cu privire la legătura dintre tipul regulatorului, parametrii săi și performanța de ansamblu obținută.

III.3.1.2 Arhitectura de control

Distanța $d_i(t)$ dintre al i -lea robot mobil și al $i-1$ -lea robot mobil poate fi calculată folosind

$$d_i(t) = x_{i-1}(t) - x_i(t). \quad (\text{III.2})$$

unde $x_i(t)$ este poziția absolută a robotului al i -lea și $x_{i-1}(t)$ este poziția absolută a robotului al $i-1$ -lea.

Sistemul de control a distanței $d_i(t)$ dintre doi roboți mobili adiacenți are o structură în cascadă, bucla interioară controlând viteza celui de-al i -lea robot și bucla exterioară pentru controlul distanței dintre robotul al i -lea și cel al $i-1$ -lea (vezi Fig. III.50).

Cinematica roboților cu roți acționate diferențial arată că mișcarea generală a robotului este efectul combinat dintre mișcarea laterală și cea longitudinală :

$$\begin{aligned} \dot{x}(t) &= \frac{R}{2} (\omega_l(t) + \omega_r(t)) \cos \theta(t), \\ \dot{y}(t) &= \frac{R}{2} (\omega_l(t) + \omega_r(t)) \sin \theta(t), \\ \dot{\theta}(t) &= \frac{R}{L} (\omega_l(t) - \omega_r(t)), \end{aligned} \quad (\text{III.3})$$

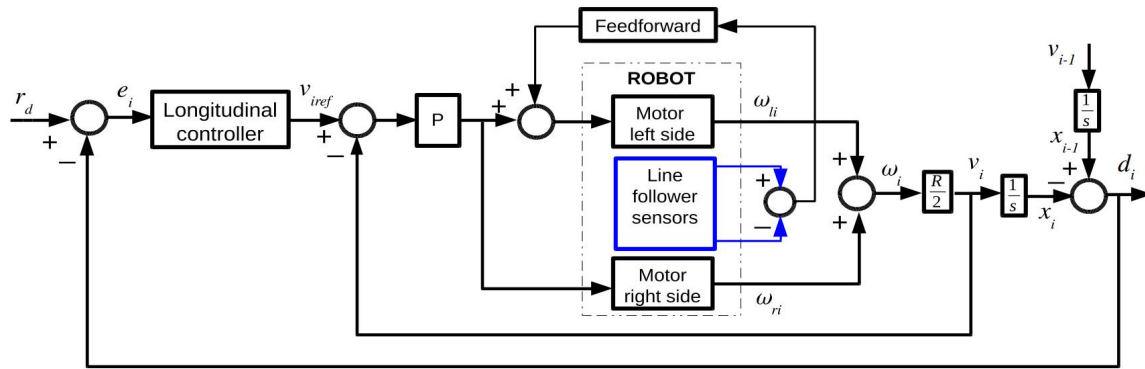


Figura III.50: Schemă de control.

unde x și y sunt poziția robotului longitudinală și respectiv laterală, R este raza roții, L este lungimea robotului, $\omega_l(t)$ și $\omega_r(t)$ sunt vitezele unghiulare pentru cele două (stâng și drept) motoare diferențiale conectate la cele două roți și $\theta(t)$ este unghiul dintre orientarea robotului și linia care trebuie urmărită. Cele două roți motoare asigură atât controlul poziției, i.e., longitudinale $x(t)$ și laterale $y(t)$, cât și orientarea, i.e., $\theta(t)$, prin vitezele unghiulare ale celor două motoare. Dacă cele două viteze unghiulare sunt egale, i.e., $\omega_l(t) = \omega_r(t)$, și robotul este perfect aliniat cu linia de la începutul experimentului, i.e., $\theta_0(t) = 0$, atunci robotul va urmări o traiectorie în linie dreaptă ($\theta(t) = 0 \Rightarrow \cos \theta(t) = 1, \sin \theta(t) = 0, \dot{\theta}(t) = 0$). Dacă traiectoria ce trebuie urmată este curbată sau condiția inițială pentru unghi este diferită de 0, i.e., $\theta_0(t) \neq 0$ sau dacă apar perturbații externe (care sunt frecvente în sistemele de timp real), atunci roboții mobili mOway au implementat un sistem de urmărire, direct în arhitectura hardware, bazat pe senzori care detectează deviația față de traiectorie și care acționează asupra motorului stâng printr-o acțiune feed-forward, crescând sau scăzând viteza motorului (vezi Fig. III.50) astfel încât robotul este menținut pe traiectoria dorită.

Din punctul de vedere al cercetării realizate în această secțiune, s-a considerat cazul în care robotul poate urmări o linie, considerând aceleași referințe pentru vitezele unghiulare ale celor două motoare (vezi Fig. III.50). Dacă apare o deviație de la referința impusă, aceasta este considerată ca o perturbație [276] care este rejectată modificând viteza unghiulară a motorului stâng printr-o acțiune feed-forward. Astfel, se consideră perturbația rejectată și atenția este îndreptată către mișcarea longitudinală a roboților.

Roboții mobili mOway au integrat în bucla interioară controlul vitezei, realizat printr-un control proporțional cu reacție negativă (vezi Fig. III.50), care are următoarea reprezentare în buclă închisă

$$G_{oi}(s) = \frac{K}{Ts + 1} = \frac{V_i(s)}{V_{iref}(s)}, \quad (III.4)$$

unde K și respectiv T sunt parametri buclei interioare, $v_i(t) = \dot{x}_i(t)$ este viteza robotului mobil și $v_{iref}(t) = u_i(t)$ este comanda dată de bucla exterioară, i.e., regulatorul (longitudinal). $V_i(s)$ și $V_{iref}(s)$ sunt transformările Laplace pentru $v_i(t)$ și respectiv $v_{iref}(t)$.

Astfel, procesul corespunzător regulatorului din bucla exterioară este descris în domeniul

timp de următoarele relații

$$\begin{aligned} T\dot{v}_i(t) + v_i(t) &= Ku_i(t), \\ \dot{x}_i(t) &= v_i(t). \end{aligned} \quad (\text{III.5})$$

Eroarea de distanță dintre al i -lea robot mobil și al $i - 1$ -lea robot din pluton poate fi definită ca:

$$e_i(t) = d_i(t) - r_d(t) = x_{i-1}(t) - x_i(t) - r_d(t), \quad (\text{III.6})$$

unde $r_d(t)$ este *referința de distanță* dorită a fi menținută între roboți, care este aceeași pentru toți roboții.

Având eroarea de distanță (III.6) definită anterior, se pot calcula următoarele relații în care variabila de timp este omisă pentru simplificare,

$$\begin{aligned} \dot{e}_i &= v_{i-1} - v_i \Rightarrow v_i = v_{i-1} - \dot{e}_i, \\ \ddot{e}_i &= \dot{v}_{i-1} - \dot{v}_i \Rightarrow \dot{v}_i = \dot{v}_{i-1} - \ddot{e}_i. \end{aligned} \quad (\text{III.7})$$

înlocuind v_i și \dot{v}_i din (III.7) în (III.5) se obține

$$\begin{aligned} T(\dot{v}_{i-1} - \ddot{e}_i) + (v_{i-1} - \dot{e}_i) &= Ku_i \Rightarrow \\ \Rightarrow T\ddot{e}_i + \dot{e}_i &= K(u_{i-1} - u_i). \end{aligned} \quad (\text{III.8})$$

Regulatorul longitudinal (vezi Fig. III.50) trebuie să includă o integrală dublă pe eroare deoarece se presupune că viteza robotului predecesor acționează ca o perturbație ce poate fi modelată ca un semnal de tip rampă la intrare. Rezultă că robotul controlat poate urmări robotul predecesor chiar și atunci când apar evenimente ca accelerări sau decelerări mari. Necesitatea unui dublu integrator a făcut imposibilă folosirea reguletoarelor PID. Considerând asta, s-au proiectat două reguletoare longitudinale predictive cu dublu integrator. Modelul robotului i pentru regulatorul longitudinal s-a obținut folosind relațiile din (III.5) și adăugând un dublu integrator pentru componenta erorii de control:

$$\begin{aligned} \dot{z}_1 &= -z_2 + v_{i-1}, \\ \dot{z}_2 &= -\frac{1}{T}z_2 + \frac{K}{T}u_i, \\ \dot{z}_3 &= -z_1 + r_d, \\ \dot{z}_4 &= z_3, \end{aligned} \quad (\text{III.9})$$

unde $z_1 = d_i$, $z_2 = v_i$, z_3 este integrala erorii de control și z_4 este dubla integrală a erorii de control.

III.3.1.3 Proiectarea regulatorului predictiv în spațiul stărilor (SS-MPC)

Fie un sistem liniar discret corespunzător modelului mișcării longitudinale a robotului mobil (III.9)

$$\mathbf{z}(k+1) = \mathbf{A}_d\mathbf{z}(k) + \mathbf{B}_d u(k), \quad k \in \mathbb{Z}_+, \quad (\text{III.10})$$

unde $\mathbf{z}(k) \in \mathbb{R}^4$ este vectorul de stare și $u(k) \in \mathbb{R}$ este intrarea controlată (i.e., referința pentru viteza robotului mobil) la eșantionul de timp k , iar \mathbf{A}_d și \mathbf{B}_d sunt matricile discretizate a sistemului.

Starea predictată în formă matriceală din [101, 277] este dată de

$$\hat{\mathbf{Z}}(k) = \mathbf{M}\mathbf{z}(k) + \mathbf{C}\mathbf{U}(k), \quad (\text{III.11})$$

unde

$$\mathbf{U}(k) := \begin{pmatrix} u(k|k) \\ u(k+1|k) \\ \vdots \\ u(k+hp-1|k) \end{pmatrix}, \quad (\text{III.12})$$

$$\hat{\mathbf{Z}}(k) := \begin{pmatrix} \hat{\mathbf{z}}(k+1|k) \\ \hat{\mathbf{z}}(k+2|k) \\ \vdots \\ \hat{\mathbf{z}}(k+hp|k) \end{pmatrix},$$

$u(k+i|k)$, $i = 0, \dots, hp-1$, este secvența viitoare de control cu hp orizontul de predicție, $\hat{\mathbf{z}}(k+i|k)$ este vectorul de stare predictat

$$\begin{aligned} \hat{\mathbf{z}}(k+i+1|k) &= \\ &= \mathbf{A}_d \hat{\mathbf{z}}(k+i|k) + \mathbf{B}_d u(k+i|k), i = 0, 1, 2, \dots \end{aligned} \quad (\text{III.13})$$

cu condiția inițială definită de $\hat{\mathbf{z}}(k|k) = \mathbf{z}(k)$,

$$\mathbf{M} = \begin{bmatrix} \mathbf{A}_d \\ \mathbf{A}_d^2 \\ \vdots \\ \mathbf{A}_d^{hp} \end{bmatrix}, \quad (\text{III.14})$$

$$\mathbf{C} = \begin{bmatrix} \mathbf{B}_d & \mathbf{0} & \dots & \mathbf{0} \\ \mathbf{A}_d \mathbf{B}_d & \mathbf{B}_d & \dots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}_d^{hp-1} \mathbf{B}_d & \mathbf{A}_d^{hp-2} \mathbf{B}_d & \dots & \mathbf{B}_d \end{bmatrix}.$$

Legea de control predictivă este calculată ca

$$\mathbf{U}^*(k) = \arg \min_{\mathbf{U}} J(k) \quad (\text{III.15})$$

fiind bazată pe minimizarea unei funcții cost cu rmătoarea formă matriceală

$$J(k) = u_0^T(k) \mathbf{H} u_0(k) + 2\mathbf{z}_0^T(k) \mathbf{F}^T \mathbf{z}_0(k) + \mathbf{z}_0^T(k) \mathbf{G} \mathbf{z}_0(k), \quad (\text{III.16})$$

unde $u_0(k) = u(k) - u_{ss}(k)$, $\mathbf{z}_0(k) = \mathbf{z}(k) - \mathbf{z}_{ss}(k)$, cu $u_{ss}(k)$ și $\mathbf{z}_{ss}(k)$ fiind valorile de regim staționar calculate din referința impusă pentru distanță, la momentul de timp discret

k și $\mathbf{H} = \mathbf{C}^T \tilde{\mathbf{Q}} \mathbf{C} + \tilde{\mathbf{R}}$, $\mathbf{F} = \mathbf{C}^T \tilde{\mathbf{Q}} \mathbf{M}$, $\mathbf{G} = \mathbf{M}^T \tilde{\mathbf{Q}} \mathbf{M} + \mathbf{Q}$, cu

$$\tilde{\mathbf{Q}} = \begin{bmatrix} \mathbf{Q} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \ddots & & \vdots \\ \vdots & & \mathbf{Q} & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \bar{\mathbf{Q}} \end{bmatrix}, \tilde{\mathbf{R}} = \begin{bmatrix} \mathbf{R} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \ddots & & \vdots \\ \vdots & & \mathbf{R} & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{R} \end{bmatrix} \quad (\text{III.17})$$

unde \mathbf{Q} , \mathbf{R} și $\bar{\mathbf{Q}}$ sunt matrici pozitiv definite (matricea \mathbf{Q} poate fi pozitiv semi-definită) cu dimensiuni adecvate.

III.3.1.4 Strategia EPSAC

Implementarea strategiei EPSAC este bazată pe tehnici de filtrare, pornind de la un model al procesului în reprezentare intrere-ieșire [278] obținut din (III.9) prin considerarea z_1 ca ieșire

$$\gamma(k) = \xi(k) + \eta(k), \quad (\text{III.18})$$

cu $\gamma(k)$, $\xi(k)$ și $\eta(k)$ ca ieșirea procesului, ieșirea modelului și respectiv perturbația proces/model. Ideea fundamentală este bazată pe predicția în cadrul modelului de bază a procesului dată de

$$\gamma(k+i|k) = \xi(k+i|k) + \eta(k+i|k), \quad (\text{III.19})$$

unde $\gamma(k+i|k)$ este predicția ieșirii la momentul discret k , în cadrul orizontului de predicție hp , bazat pe măsurători anterioare și valori postulate pentru intrări. Predicția ieșirii modelului $\xi(\cdot|\cdot)$ și a zgomotului colorat $\eta(\cdot|\cdot)$ pot fi obținute recursiv din modelul procesului și respectiv din tehnici de filtrare. Zgomotul colorat poate fi modelat ca

$$\eta(k) = \frac{C(q^{-1})}{D(q^{-1})} \varepsilon(k), \quad (\text{III.20})$$

cu $C(q^{-1})$ și $D(q^{-1})$ polinoame în operatorul q^{-1} și $\varepsilon(k)$ ca zgomot alb, fiind proiectat pentru robustețe întruiva diferențelor proces-model. Controlul optimal este obținut prin minimizarea următoarei funcții de cost

$$J(k) = \sum_{i=hm}^{hp} [r(k+i|k) - \gamma(k+i|k)]^2 + \sum_{i=0}^{hu-1} [\Delta u(k+i|k)]^2, \quad (\text{III.21})$$

cu $r(\cdot|\cdot)$ fiind referința de distanță dorită. hm este minimul orizontului de predicție calculat de obicei ca timpul mort al procesului+1 și incrementele comenzii viitoare $\Delta u(\cdot|\cdot)$ pot varia în cadrul orizontului de control hu .

în absența restricțiilor, soluția problemei de optimizare (III.43) poate fi obținută offline $\mathbf{U}^*(k) = -\mathbf{H}^{-1} \mathbf{F} \mathbf{z}(k)$ și conform cu principiul orizontului alunecător [101,277] doar primul element din secvența de comandă \mathbf{U}^* este aplicat procesului, fiind definit ca

$$u(k) = u^*(k|k) = \mathbf{K}_{hp} \mathbf{z}(k), \quad (\text{III.22})$$

unde $K_{hp} = - [I_{hp} \ 0 \ \dots \ 0] H^{-1}F$ este factorul de amplificare a legii de reglare predictive.

III.3.1.5 Roboții mobili mOway

Roboții mobili mOway sunt foarte folositori în demonstrarea teoriilor relaționate cu problema controlului unui pluton deoarece au un cost de achiziție scăzut și pot fi programați ușor.

Descrierea funcțională Robotul mOway folosit în această secțiune este un robot cu acționare diferențială pe roți, aceste tipuri de roboți fiind folosiți de obicei în aplicații multi-robot, datorită manevrării lor rapide, costului scăzut și simplității [279]. Robotul mOway este controlat de un microcontroler PIC18F87J50 cu frecvența de 4 MHz, și folosește un grup de servo-motoare pentru fiecare din cele două roți. De-asemenea are integrată o buclă pentru controlul vitezei, bazată pe o reglare proporțională cu reacție negativă primită de la semnalul codificatoarelor. Rotația roții este monitorizată de o etichetă de codificare și un senzor infraroșu. Atunci când eticheta este poziționată pe segmentul negru, ieșirea logică este egală cu 1 și când este poziționată pe sectorul alb, ieșirea este egală cu 0. Microcontroler-ul analizează aceste semnale folosind un algoritm intern care nu este disponibil utilizatorului (ce poate determina poziția exactă a roții prin măsurarea unui impuls cu o precizie de ± 1.7 mm) și acționează asupra motoarelor. În acest fel, robotul mOway este capabil să își mențină viteza constantă indiferent de suprafață.

Robotul mOway folosește senzori de detecție a obstacolelor bazați pe o lumină infraroșie, pentru a detecta obiecte localizate în fața sa; acești senzori sunt de-asemenea folosiți pentru a măsura distanța dintre robot și cel din față. Senzorul funcționează astfel: emițătorul de lumină generează un puls de 70 us care permite receptorului să captureze orice obstacol folosind o etapa de filtrare și amplificare. După ce semnalul este procesat electronic, microcontroler-ul PIC îl poate măsura folosind un convertor analog-la-digital sau ca o intrare digitală. Distanța minimă care poate fi măsurată este aproape 3 cm și cea maximă este în jur de 11 cm. Senzorii pentru detecția obstacolelor sunt foarte sensitivi față de culoarea podelei și a luminozității mediului (se recomandă un mediu luminos pentru a mări refracția luminii infraroșie) și astfel perturbațiile sunt introduse în sistem.

Totodată, robotul mOway este echipat cu doi senzori pentru urmărirea liniei (Vishay CNY70) montați în partea frontală joasă a robotului. Cei doi senzori de urmărire a liniei sunt folosiți pentru a urmări un circuit marcat pe podea în scopul testării și validării arhitecturii de control proiectată, alcătuit dintr-o bandă neagră (senzorii înregistrează un voltaj ridicat) pe o suprafață albă (voltaj scăzut) astfel încât senzorii pot identifica cu ușurință culoarea liniei.

Modelarea robotului mOway Modelul robotului este determinat folosind un experiment de identificare alcătuit dintr-o secvență de semnale treaptă, ilustrat în Fig. III.51. Viteza robotului mobil este controlată intern, astfel încât dinamica robotului poate fi reprezentată de o funcție de transfer de ordinul întâi, deoarece robotul este un proces simplu bazat pe motoare de curent continuu și un regulator (proporțional) P.

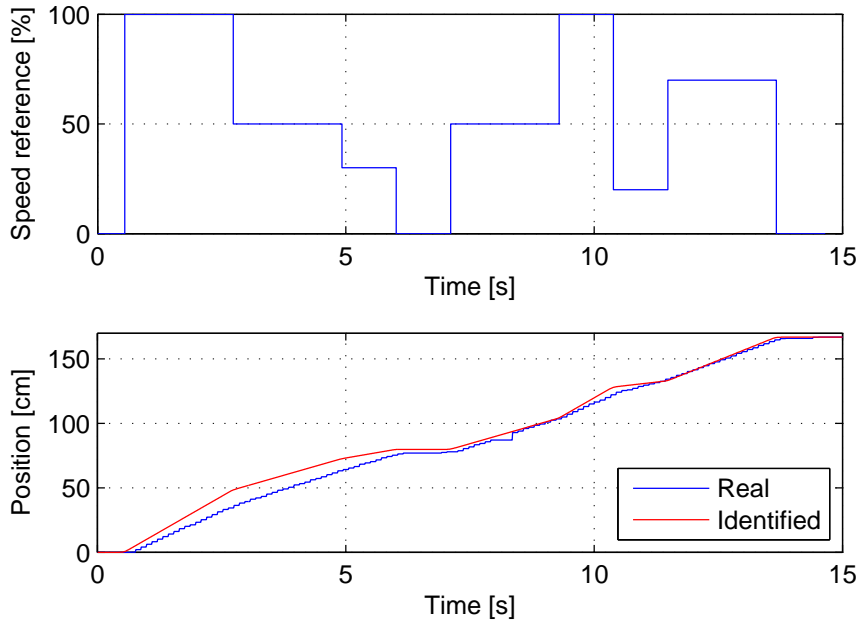


Figura III.51: Experimentul de identificare.

Funcția de transfer rezultantă dintre viteza de referință și poziția robotului este dată de un sistem de ordinul întâi plus un integrator

$$G_o(s) = \frac{1}{s} G_{oi}(s) = \frac{K}{s(Ts + 1)} = \frac{0.22}{s(0.04s + 1)} \quad (\text{III.23})$$

Comparația dintre ieșirea reală a robotului (i.e., poziția) și ieșirea modelului identificat este ilustrată în Fig. III.51, similitudinea dintre cele două semnale fiind de 94%. Rețineți că intrarea procesului nu este o forță, ci o referință pentru viteza robotului dată în procente. În același timp, viteza este măsurată în cm/s și are un maxim de 22 cm/s, obținându-se un factor de amplificare de 0.22 pentru funcția de transfer.

Perioada de eșantionare a fost aleasă $T_s = 5$ ms, din raționamente bazate pe dinamica sistemului (constanta de timp a sistemului este de 40 ms).

III.3.1.6 Rezultate obținute

Formația predefinită a plutonului de roboți mobili are o structură în lanț, ceea ce înseamnă că fiecare robot mobil este un urmăritor pentru robotul precedent și un lider pentru robotul succesiv. În testele de simulare și experimentale efectuate, toți roboții mobili urmăresc o linie marcată pe podea. Plutonul este alcătuit din roboți mobili identici, i.e., un pluton omogen, în care toți membrii plutonului au aceeași structură hardware și software, și fiecare robot mobil are informații cu privire la distanța dintre el și robotul din față. În această secțiune, s-a considerat că toți roboții mobili din pluton sunt controlați folosind un sistem de control de croazieră adaptiv (eng. adaptive cruise control) (ACC) (referința pentru sistemul în buclă închisă fiind o distanță predefinită dinterobotul curent și cel din față sa), cu excepția robotului lider al plutonului, care este reglat cu un sistem de control de croazieră

(eng. cruise control) (CC) (referința pentru sistemul în buclă închisă fiind o viteză predefinită pentru robotul mobil). În scopul simulărilor s-a implementat un pluton de 17 roboți mobili (1 lider și 16 urmăritori), în timp ce rezultatele experimentale au fost obținute pentru un pluton de 4 roboți mobili (1 lider și 3 urmăritori).

S-a efectuat un test lung, însă doar un segment de timp ilustrativ atât pentru simulare, cât și pentru experimente este prezentat; în acest segment de timp apar mai multe schimbări de referință pentru viteza liderului. Distanța inițială dintre toți roboții din pluton și distanța dorită sunt de 7 cm.

Regulatele ACC au fost proiectate pe baza unui model simplu al robotului mobil, preluat din [19, 267]

$$X(s) = G_o(s)R_s(s), \quad (\text{III.24})$$

unde $X(s)$ este transformata Laplace pentru $x(t)$ și $R_s(s)$ este transformata Laplace pentru $r_s(t)$, care este referința de viteză a robotului mobil.

III.3.1.7 Stabilitate

Stabilitatea sirului de roboți mobili ca alcatuiesc plutonul este una dintre cele mai importante aspecte de considerat atunci când se proiectează regulate longitudinale [10, 270] spre a evita efectele adverse negative asupra traficului și a siguranței. În continuare se presupune că toți roboții mobili din pluton sunt identici. Aceasta ipoteză simplificatoare este în concordanță cu aplicația propusă (controlul unui număr de roboți mobili mOway identici). Totuși, plutoanele realiste pot fi alcătuite din roboți mobili diferiți, cu dinamici diferite ceea ce înseamnă că regulatorul fiecărui robot mobil trebuie să fie proiectat în funcție de dinamica specifică.

Pentru a asigura stabilitatea sirului de roboți mobili, la fiecare moment de timp trebuie îndeplinite două condiții [10, 31, 280]:

- eroarea de regim tranzitoriu nu trebuie să se amplifice (eroarea de distanță trebuie să fie atenuată) începând de la lider către ultimul robot din pluton;
- valorile absolute ale accelerațiilor trebuie să descrească spre finalul plutonului.

Următoarea condiție trebuie să fie îndeplinită, spre a asigura stabilitatea sirului de roboți mobili

$$\|e_1\|_\infty \geq \|e_2\|_\infty \geq \dots \geq \|e_N\|_\infty, \quad (\text{III.25})$$

unde $\|e_i\|_\infty = \sup_t |e_i(t)|$ cu $i = 1, \dots, N$, unde N este numărul de roboți mobili din pluton.

Pentru doi roboți mobili adiacenți, funcția de transfer pentru eroarea de distanță poate fi definită ca

$$G(s) = \frac{E_i(s)}{E_{i-1}(s)}, \quad (\text{III.26})$$

unde $E_i(s)$ și $E_{i-1}(s)$ sunt transformatele Laplace ale $e_i(t)$ și respectiv $e_{i-1}(t)$. Conform teoriei sistemelor liniare [281] este cunoscut faptul că $\|e_i\|_\infty \leq \|g\|_1 \|e_{i-1}\|_\infty$, unde $g(t)$ este inversa transformatei Laplace pentru $G(s)$ și $\|g\|_1 = \int_0^\infty |g(t)| dt$. Așadar, o condiție suficientă pentru a atenua eroarea de distanță și a garanta că perturbațiile nu se amplifică

prin propagarea în pluton [280, 282] este ca norma L_∞ -indusă a operatorului liniar core-spunzător să fie mai mică decât unu, i.e., ca $\|g\|_1 \leq 1$.
 În plus, este cunoscut [281] că $|G(0)| \leq \|G\|_\infty \leq \|g\|_1$, unde $\|G\|_\infty = \max_\omega |G(j\omega)|$ și $j^2 = -1$. Folosind definiția transformatei Laplace se poate obține $|G(0)| = \left| \int_0^\infty g(t) dt \right| \leq \int_0^\infty |g(t)| dt = \|g\|_1$.

Dacă răspunsul la impuls $g(t)$ rămâne pozitiv, atunci

$$\|G\|_\infty = |G(0)| = \|g\|_1 \quad (\text{III.27})$$

și condiția $|G(0)| \leq 1$ asigură că stabilitatea șirului pentru plutonul de roboți mobili L_∞ . Acesta este singurul caz în care norma L_1 a $g(t)$ poate fi evaluată în domeniul frecvență. Dacă răspunsul la impuls $g(t)$ își schimbă semnul în timp, atunci

$$\|G\|_\infty \leq 1, g(t) \geq 0, \forall t \quad (\text{III.28})$$

garantează doar stabilitatea șirului pentru plutonul de roboți mobili în sensul normei L_2 , i.e., $\|e_i\|_2 \leq \|G\|_\infty \|e_{i-1}\|_2$, unde $\|e_i\|_2 = \sqrt{\int_0^\infty e_i(t)^2 dt}$. Pentru a asigura siguranța, se dorește a avea $\|g\|_1 \leq 1$, dar, din nefericire, garantarea semnelui pentru $g(t)$ nu este întotdeauna computațional simplă.

În cele ce urmează, se vor proiecta două regulatoare P în scopul comparației.

Condiția pozitivității răspunsului la impuls $g(t)$ trebuie să fie verificată pentru a putea analiza stabilitatea șirului plutonului de roboți mobili cu regulator P de forma $C_i(s) = K_{P_i}$, $i = 1, \dots, N$. Considerând regulatoare proporționale, i.e., $u_i = K_{P_i} e_i$ și $u_{i-1} = K_{P_{i-1}} e_{i-1}$, în (III.8), funcția de transfer pentru eroarea de distanță (III.26) devine

$$G(s) = \frac{K K_{P_{i-1}}}{T s^2 + s + K K_{P_i}} \quad (\text{III.29})$$

În continuare se presupune că regulatoarele P sunt identice pentru toți roboții, i.e., $K_{P_{i-1}} = K_{P_i} = K_P$. Stabilitatea Intrare-Mărginită-Ieșire-Mărginită (eng. Bounded-Input Bounded-Output) (BIBO) ¹ pentru fiecare robot mobil din pluton este întotdeauna îndeplinită dacă următoarele inegalități sunt satisfăcute

$$T > 0, K > 0 \text{ and } K_P > 0. \quad (\text{III.30})$$

Condiția necesară pentru a obține un răspuns la impuls pozitiv, dacă sistemul nu are zerouri este ca polii sistemului să fie reali, ceea ce duce la următoarea restricție pentru alegerea factorului de amplificare a regulatorului

$$K_P \leq \frac{1}{4KT}. \quad (\text{III.31})$$

Dacă această condiție este satisfăcută, atunci condiția pentru stabilitatea șirului L_∞ din (III.27) poate fi verificată

$$|G(0)| \leq 1 \Leftrightarrow \left| \frac{K K_P}{K K_P} \right| \leq 1 \quad (\text{III.32})$$

¹Definiție: Un sistem este BIBO stabil dacă fiecare intrare mărginită produce o ieșire mărginită [283] [Capitolul 14].

care este adevărată pentru toate valorile K_P .

Dacă răspunsul la impuls $g(t)$ își schimbă semnul ($K_P > \frac{1}{4KT}$) poate fi analizată stabilitatea șirului mai puțin restrictivă L_2 folosind inegalitatea (III.28) care duce la următoarea restricție pentru alegerea parametrului regulatorului

$$K_P \leq \frac{1}{2KT}. \quad (\text{III.33})$$

III.3.1.8 Rezultate obținute în simulare

Rezultatele obținute cu regulatorul P în care parametrul regulatorului a fost ales conform condiției din (III.31) și folosind modelul din (III.23) care determină valoarea $K_P \leq 28.4$ sunt ilustrate în Fig. III.52 pentru $K_P = 25$. Deși plutonul de roboți mobili este stabil L_∞ performanțele roboților mobili individuali nu sunt foarte bune, având eoare de regim staționar în distanțe.

Rezultatele obținute cu regulatorul P în care parametrul regulatorului a fost ales conform condiției din (III.33) și folosind modelul din (III.23) care determină valoarea $K_P \leq 56.8$ sunt ilustrate în Fig. III.53 pentru $K_P = 50$. Deși plutonul de roboți mobili este stabil L_2 , performanțele roboților mobili individuali nu sunt foarte bune, având eoare de regim staționar în distanțe, dar aceste erori staționare sunt mai mici decât cele obținute cu regulatorul anterior.

Rezultatele experimentale obținute cu cele două regulatoare P cu $K_P = 25$ și $K_P = 50$ sunt ilustrate în Fig. III.54 și respectiv Fig. III.55. Poate fi observat că creșterea factorului K_P conduce la un răspuns mai rapid cu erori staționare mai mici, dar cu oscilații mari. Rețineți că răspunsurile oscilante sunt cauzate și de efectele perturbațiilor asupra măsurătorilor sensorului de distanță. De-aseamenea, poate fi observat că erorile de regim staționar obținute cu regulatoarele proporționale sunt mai mari decât cele obținute cu regulatoarele predictive.

III.3.1.9 Rezultate experimentale

Modelul discret (III.36) este obținut prin discretizarea modelului (III.9) cu parametri din (III.23) și perioada de eșantionare $T_s = 5$ ms. După cum a fost menționat în secțiunea III.3.1.5, perioada de eșantionare a fost aleasă în funcție de dinamica sistemului (constante de timp a sistemului este 40 ms).

Rezultatele obținute cu regulatorul MPC proiectat în spațiul stărilor, pentru care parametri de acord au fost aleși astfel încât s-a obținut un compromis între performanțe bune și o buclă închisă rapidă, cu orizontul de predicție $hp = 5$ sunt ilustrate în Fig. III.66. Rețineți că a doua stare (viteza robotului) nu a fost măsurată, ci calculată ca derivata primei stări.

Rezultatele obținute cu regulatorul EPSAC, cu următorii parametri de acord: $hm = 1$ (nu există timp mort în proces), $hp = 2$ (o valoare mai mică produce o buclă închisă mai rapidă), $hu = 1$ (orizontul de control implicit), $C(q^{-1}) = 1$ și $D(q^{-1}) = 1 - q^{-1}$ sunt ilustrate în Fig. III.57.

Rezultatele experimentale obținute cu aceleași regulatoare SS-MPC and EPSAC sunt reprezentate în Figurile III.67 și III.59, în care se poate vedea ca eroarea de distanță este aproape de 0. Rețineți ca răspunsurile oscilante sunt cauzate și de efectele perturbațiilor asupra măsurătorilor sensorului de distanță. Adițional, se observă ca vitezele urmăriturilor sunt

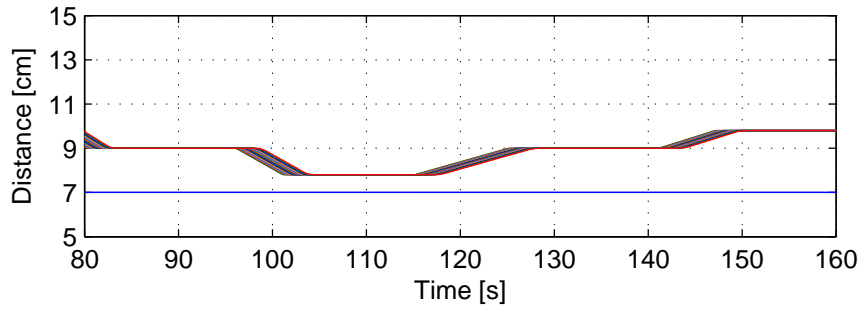
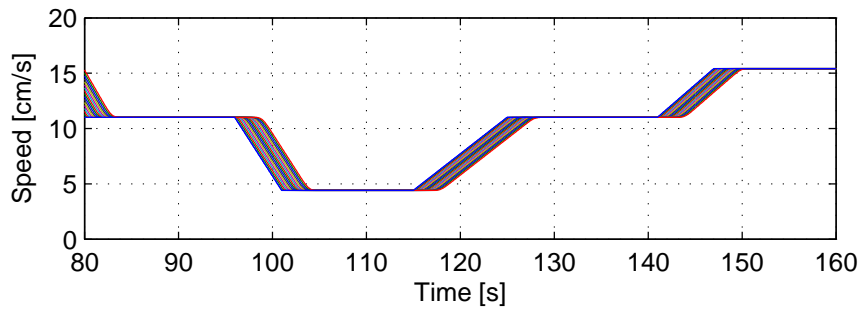


Figura III.52: Simulare: vitezele și distanțele roboților mobili cu regulator P (L_∞).

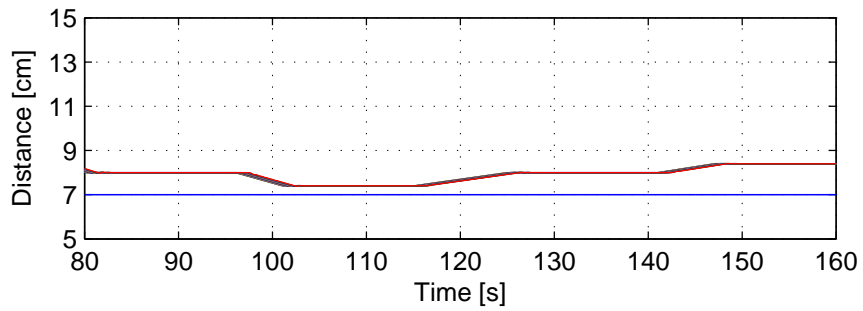
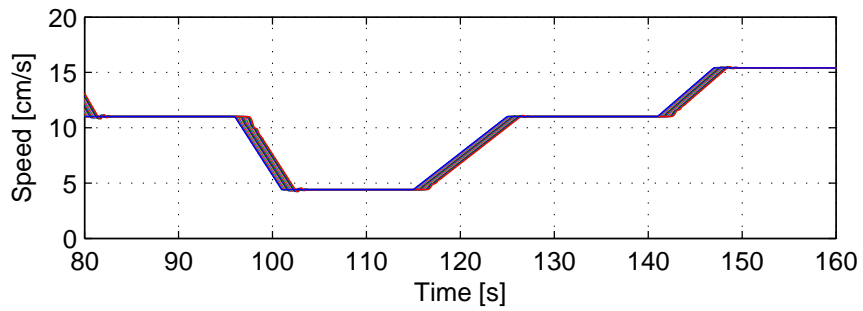


Figura III.53: Simulare: vitezele și distanțele roboților mobili cu regulator P (L_2).

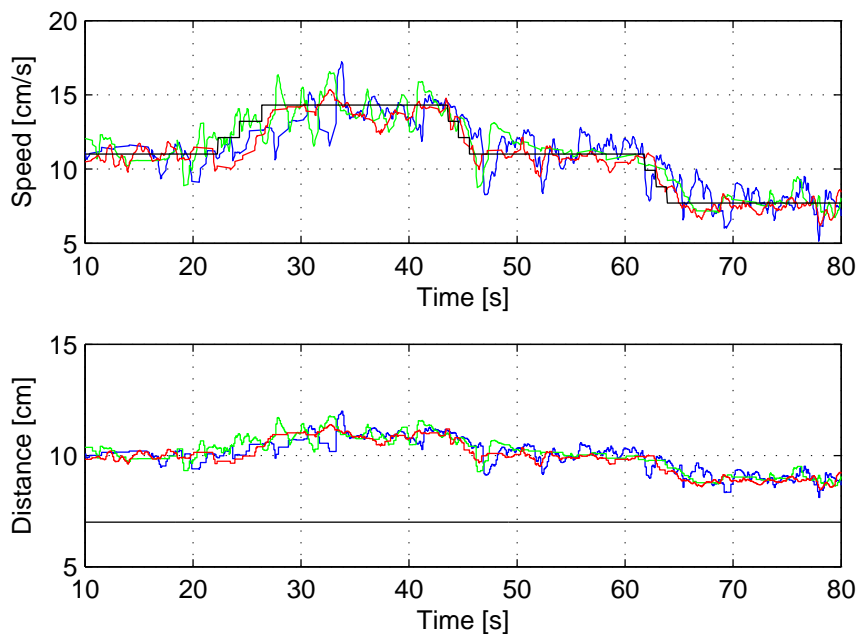


Figura III.54: Experiment: vitezele și distanțele roboților mobili cu regulator P (L_∞).

mai mari decât cea a liderului, deoarece urmăritorii trebuie să se deplaseze mai mult pentru a menține referința de distanță predefinită față de robotul din față, ceea ce implică ca să atinge o viteză medie mai mare (vezi secțiunea III.3.2.5 pentru detalii suplimentare)

III.3.1.10 Remarci experimentale

Se poate observa că în Figurile III.67 și III.59 vitezele medii ale urmăritorilor sunt în mod continuu mai mari decât viteza liderului. Aceasta se datorează câtorva circumstanțe privitoare la configurația experimentului:

- în experimentele efectuate toți roboții mobili urmăresc o linie marcată pe podea;
- roboții mobili nu rămân perfect pe linie în timpul experimentului, dar, după cum s-a explicat în secțiunea III.3.1.5, dacă se produce o deviație de la referința impusă, aceasta este considerată ca o perturbație [276] care este rejectată prin modificarea vitezei unghiulare a motorului stâng printr-o acțiune feed-forward *gain*; totuși, aceasta înseamnă că roboții trebuie să se deplaseze mai mult pentru a menține traiectoria de distanță predefinită, ceea ce implică ca să obțină o viteză medie mai mare;
- mișcarea în jurul liniei ce trebuie urmărită influențează distanța măsurată de fiecare robot (dacă liderul este drept pe linie și urmăritorul este orientat un pic mai la stânga/dreapta atunci distanța măsurată de urmăritor este mai mare decât distanța reală, ceea ce corespunde cu o viteză medie mai mare);
- când liderul atinge o curbă, urmăritorul nu poate detecta precis distanța (de obicei distanța măsurată este mai mare) ceea ce cauzează ca viteza să crească pentru a reduce eroarea artificială;

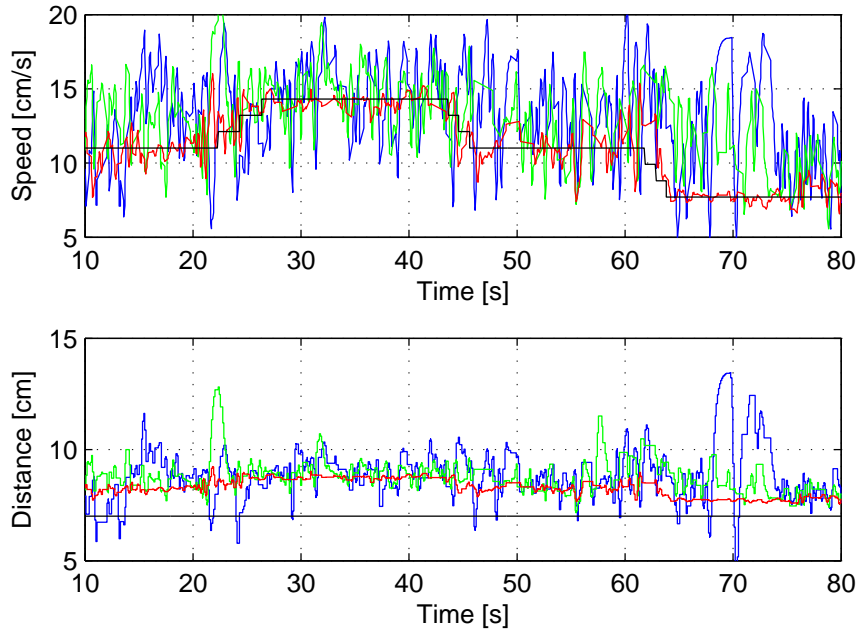


Figura III.55: Experiment: vitezele și distanțele roboților mobili cu regulator P (L_2).

- se poate observa în Figurile III.54 și III.55 că folosind regulatorul P L_∞ sau P L_2 , vitezele medii ale următorilor sunt similare cu viteza liderului deoarece aceste două regulatoare sunt mai puțin agresive, comparativ cu celelalte regulatoare (i.e., regulatoarele proportional sunt proiectate cu garanția stabilității șirului, ceea ce descrește performanța obținută, e.g., erori staționare mai mari).

III.3.1.11 Evaluarea performanței

Performanțele obținute de regulatoarele proiectate sunt evaluate și cantitativ folosind indexul caracteristic IAE care corespunde cu valoarea absolută a erorii integrale

$$IAE = \int_0^{\infty} |e(t)| dt. \quad (III.34)$$

Indexul este sumarizat în Tabelul 7 și Tabelul 8 pentru rezultatele în simulare și respectiv experimentale. Se poate observa ca valorile indexului IAE pentru regulatoarele P stabile în sensul L_∞ și L_2 , regulatoarele MPC pe stare și cele EPSAC sunt descrescătoare de la lider spre ultimul robot din pluton, ceea ce înseamnă ca erorile de distanță sunt atenuate, astfel încât plutonul de roboți este stabil. În plus, valorile obținute pentru cele două regulatoare predictive sunt mai mici decât cele obținute cu regulatoarele P.

III.3.1.12 Concluzii

În această secțiune s-a considerat proiectarea a două regulatoare MPC pentru controlul unui pluton de roboți mobili, cu considerente de stabilitatea șirului. A fost oferită o soluție

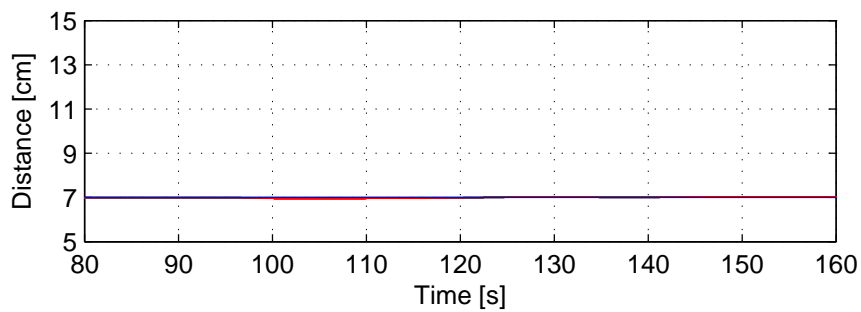
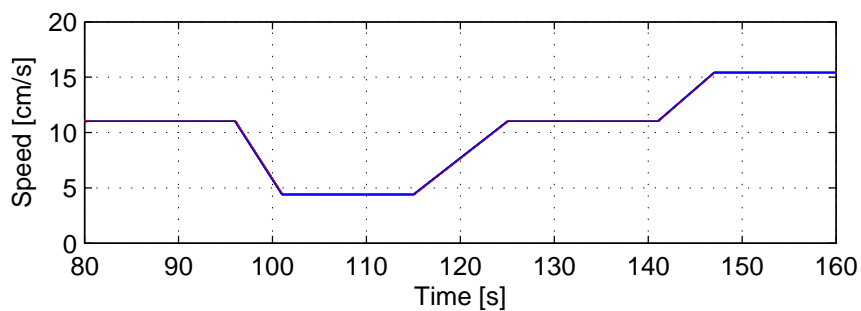


Figura III.56: Simulare: vitezele și distanțele roboților mobili cu regulator (SS-MPC).

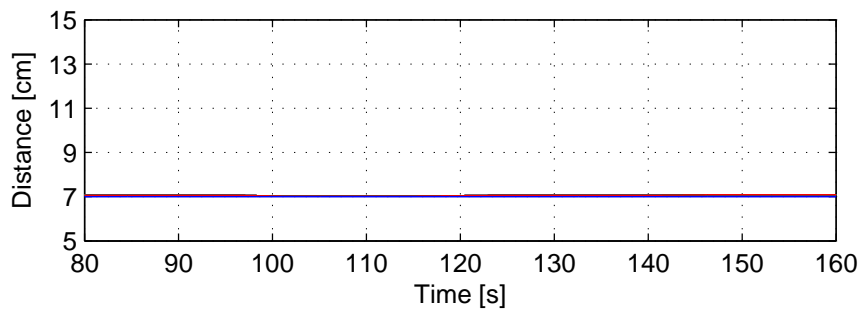
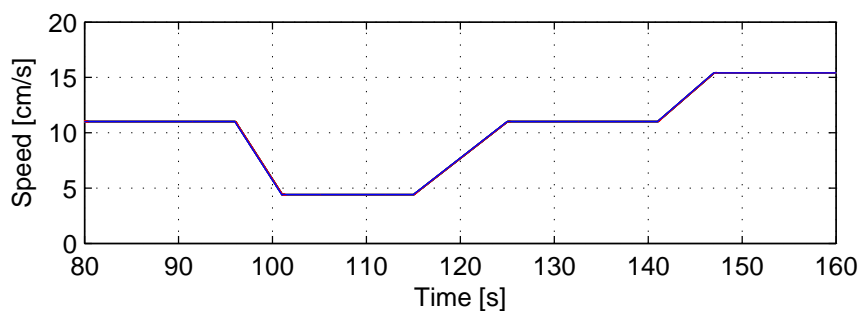


Figura III.57: Simulare: vitezele și distanțele roboților mobili cu regulator (EPSAC).

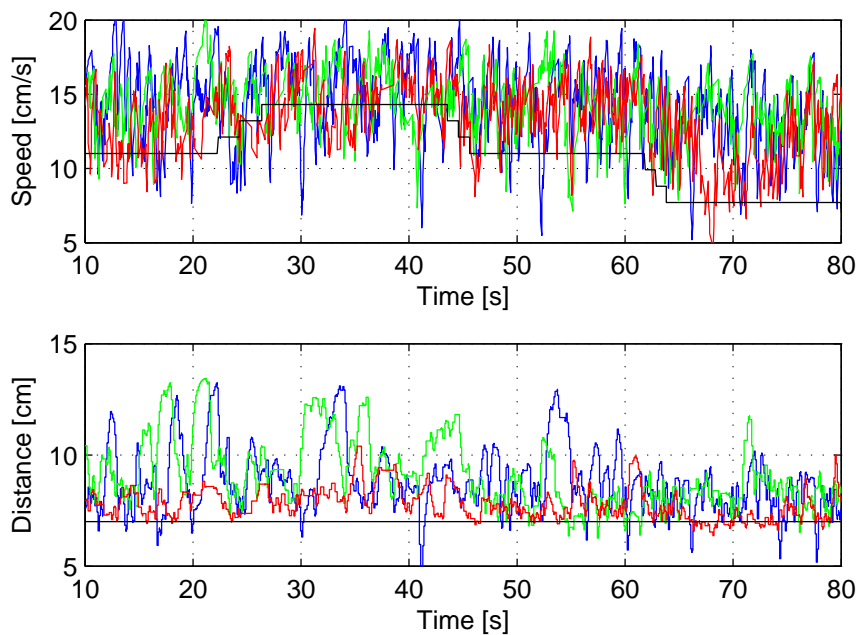


Figura III.58: Experiment: vitezele și distanțele roboților mobili cu regulator (SS-MPC).

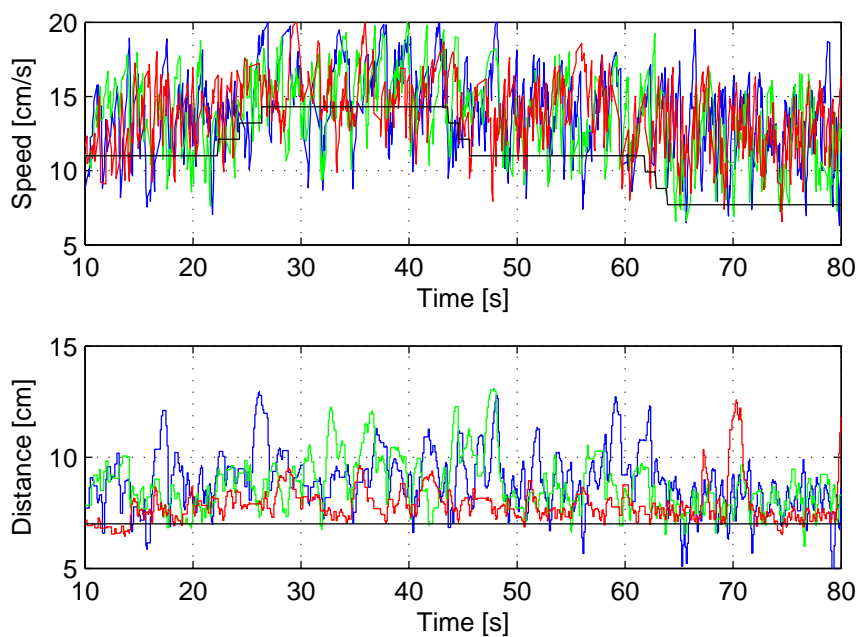


Figura III.59: Experiment: vitezele și distanțele roboților mobili cu regulator (EPSAC).

Table 7: Indexul caracteristic IAE - rezultate în simulare

| Regulator | $P(L_\infty)$ | $P(L_2)$ | SS-MPC | EPSAC |
|-----------|---------------|----------|--------|--------|
| 1 | 146.9071 | 73.4881 | 0.5237 | 2.8164 |
| 2 | 146.7732 | 73.4525 | 0.5237 | 2.8164 |
| 3 | 146.6480 | 73.4179 | 0.5237 | 2.8164 |
| 4 | 146.5317 | 73.3844 | 0.5237 | 2.8163 |
| 5 | 146.4241 | 73.3520 | 0.5237 | 2.8163 |
| 6 | 146.3235 | 73.3207 | 0.5237 | 2.8163 |
| 7 | 146.2354 | 73.2905 | 0.5237 | 2.8163 |
| 8 | 146.1542 | 73.2614 | 0.5236 | 2.8163 |
| 9 | 146.0818 | 73.2334 | 0.5236 | 2.8162 |
| 10 | 146.0182 | 73.2066 | 0.5236 | 2.8162 |
| 11 | 145.9634 | 73.1808 | 0.5236 | 2.8162 |
| 12 | 145.9174 | 73.1561 | 0.5236 | 2.8162 |
| 13 | 145.8802 | 73.1325 | 0.5236 | 2.8161 |
| 14 | 145.8518 | 73.1100 | 0.5236 | 2.8161 |
| 15 | 145.8321 | 73.0886 | 0.5236 | 2.8161 |
| 16 | 145.8213 | 73.0683 | 0.5236 | 2.8161 |

Table 8: Indexul caracteristic IAE - rezultate experimentale

| Regulator | $P(L_\infty)$ | $P(L_2)$ | SS-MPC | EPSAC |
|-----------|---------------|----------|--------|-------|
| 1 | 256 | 133 | 143 | 134 |
| 2 | 211 | 131 | 110 | 125 |
| 3 | 202 | 125 | 58 | 62 |

simplă pentru controlul unui pluton fără comunicație între roboți în care fiecare robot mobil are acces doar la distanța față de robotul din față sa. S-au comparat rezultatele cu câteva regulatoare clasice P cu garanția stabilității șirului inclusă, iar rezultatele obținute ilustrează faptul că performanțele pot fi îmbunătățite. Totodată, s-a observat ca prin asigurarea stabilității șirului cu regulatoarele P, performanțele individuale ale roboților au scăzut: cea mai mare eroare staționară a fost obținută cu regulatorul P având garanția stabilității șirului în sensul L_∞ . Totuși, performanțele obținute cu regulatoarele MPC având stabilitatea garantată au fost îmbunătățite față de cele obținute cu regulatoarele P. Analiza performanței a arătat că pentru cele doua regulatoare MPC, erorile de regim staționar sunt aproape de 0 și ca plutonul este stabil.

III.3.2 Utilizarea comunicațiilor wireless pentru colectarea datelor în plutoanele de vehicule

Dezvoltarea rapidă a dispozitivelor conectate din ziua de azi conduce la o creștere exponențială a oportunităților pentru tehnologiile din industria auto, folosind sisteme conectate. Echipând vehiculele cu tehnologii capabile să transmită informații, să primească și să proceseze date, impactul asupra siguranței, confortului și mobilității este major [284]. La momentul actual, vehiculele conectate sunt prezente în formatul vehicul cu vehicul (V2V), acestea presupunând comunicații între vehicule, vehicul cu infrastructura (V2I), presupunând comunicații între vehicul și infrastructura, și vehicul cu vehicul și infrastructura (V2X). Toate aceste tehnologii se bazează pe comunicații dedicate cu raza scurtă, care transmit și

recepționează informații în vehicul. Aplicațiile care pot face uz de aceste comunicații sunt de exemplu evitarea coliziunilor cu alte vehicule [284]. Aplicațiile V2I au scopuri mai largi, ele tinzând să fie mai puțin urgente sau mai puțin critice din punct de vedere al siguranței. În contrast cu V2V, aplicațiile V2I au nevoie doar de infrastructura și vehiculul contactat la rețeaua de comunicații [285].

Studii preliminare arată că V2I poate scădea cu 12% probabilitatea de accidente, pentru aplicații critice din punct de vedere al siguranței [286]. Viziunea aplicațiilor V2I, ca și a aplicațiilor V2V, este însă de a reduce cât mai mult accidentele în trafic. În V2I, infrastructura joacă un rol de coordonare prin generarea de informații de localizare globale sau locale, informații despre trafic și despre condițiile drumurilor, ceea ce poate duce la sugestii sau chiar cereri de comportament adecvat a vehiculelor în funcție de aceste informații. Vitezele și distanțele dintre vehicule pot fi transmise de către infrastructură pe baza condițiilor de trafic, cu scopul de a optimiza emisiile, de a reduce consumul de carburant, și de a evita blocajele în trafic. Informațiile pot fi trimise folosind afișaje plasate de-a lungul drumului sau direct către vehicul prin rețele de comunicații wireless. Mai departe, având vehiculele echipate cu sistemele de control necesare, infrastructura ar putea controla autovehiculul în mod direct.

Pe lângă acestea, sistemele de asistență ale șoferului folosind camere (mono, stereo, sau surround view) și senzori (radar, ultrasonici, lidar) sunt considerate ca o soluție pentru a crește siguranța în trafic, evitarea ambuteiajelor, și de a ușura condusul. Un sistem folosind fuziunea de senzori poate fi folosit pentru a detecta obstacole [287]. Senzorii ultrasonici, cu posibilitate de detecție ridicată și preț redus pot fi folosiți pentru detecții pe distanțe reduse. Pentru distanțe mai mari, camera și radarul pot fi folosite pentru detecția de obiecte. Pe lângă acestea, sistemul de poziționare global (GPS) și unitatea de măsurare inerțială (IMU) sunt folosite pentru a măsura poziția și orientarea vehiculului. Toate aceste informații ar putea fi trimise către infrastructura prin comunicații V2I și pot fi folosite pentru a informa alte vehicule despre orice situație de urgență.

Este dificil pentru un sistem de achiziționare de date tradițional să colecteze în mod ideal date de la senzori, într-un mediu deschis, cum ar fi autostrăzi sau drumuri publice în orașe. Senzorii statici nu pot să scaleze suficient de bine un mediu deschis, și chiar o plasare mai densă a acestor senzori statici nu poate duce la măsurători de calitate foarte înaltă, de aceea adesea se folosesc interpolări pentru a estima variațiile [288]. Odată cu introducerea vehiculelor autonome, toate aceste informații pot fi folosite pentru a colecționa date despre mediul înconjurător. Folosirea senzorilor instalați pe vehicule, pare o soluție mai bună decât utilizarea senzorilor statici, în ceea ce privește numărul de locații de unde se iau măsurători și oferă o acuratețe și fiabilitate mai bună. Mai mult decât atât, folosind GPS-ul, vehiculele se pot localiza în mod continuu și cu o precizie ridicată, în timp ce adăugarea altor dispozitive de măsurare poate îmbunătăți chiar mai mult calitatea măsurătorilor și crește șansa detecției situațiilor de urgență.

Fiecare măsurătoare poate fi văzută ca o pereche de timp și valoare $\langle t_i, s_i \rangle$. În timp ce vehiculul este în mișcare, acesta poate foarte ușor să colecționeze informațiile de la senzori, știind și timpul și locația în mediul în care se află. Știind că diferiți senzori fac măsurători la intervale diferite de timp, se poate face o aliniere a măsurătorilor $\langle t_i, s_i \rangle$ pe baza valorilor de timp stocate, astfel obținând cea mai bună estimare a locației $\langle t_i, x_i, y_i \rangle$,



Figura III.60: mOway robot

pentru fiecare măsurătoare a senzorilor $\langle t_i, x_i, y_i, s_i \rangle$.

III.3.2.1 Descrierea sistemului

Robotul considerat pentru acest studiu este un robot mobil cu roți diferențiale (Fig. III.60), un robot mic programabil autonom conceput în principal pentru a aplicații mini-robotice practice [289].

Roboții mobili mOway sunt echipați cu o gamă largă de senzori, cum ar fi senzori de temperatură, accelerometru cu 3 axe, senzor de intensitate a luminii, senzori anti-coliziune și infraroșu. Motoarele sunt controlate utilizând protocolul Inter-Integrated Circuit (I2C). Configurația robotului este ilustrată în Fig. III.61. Așa cum se poate vedea în figură, robotul este controlat de un microcontroler PIC Microchip pe 8 biți (PIC18), care funcționează cu o frecvență de 4Mhz [289] și are 4KB de RAM și 32 de conexiuni [290].

Pentru fiecare dintre cele două roți din spate este folosit un grup de servomotoare pentru a conduce robotul. Sistemul de tracțiune este ilustrat în detaliu în Fig. III.61 în dreptunghiul roșu din dreapta. Microcontrollerul secundar microchip PIC de 8 biți (PIC16) trimite comenzile I2C (adică semnalele PWM) către sistemul de acționare (de exemplu, puntea H) care controlează motoarele, în timp ce senzorul infraroșu este utilizat pentru măsurarea viteza roții. Pentru a controla viteza robotului este folosit un regulator proporțional intern cu feedback negativ de la encodere, menținând viteza constantă pe diferite suprafețe de rulare.

Robotul include doi senzori de urmărire a liniei (Vishay CNY70) montați în partea din față. Sursa de lumină care emite și detectorul sunt aranjate în aceeași direcție, ceea ce înseamnă că lumina reflectată poate fi detectată la fața locului. Există 3 tipuri de suprafețe pe care senzorul le poate detecta:

- clear → lumina infraroșie este aproape complet reflectată și senzorul înregistrează o tensiune scăzută;
- colored → reflecția fasciculului este parțială, iar senzorul va înregistra diferite nivele în funcție de culoare; în acest fel, pot fi identificate și culorile;

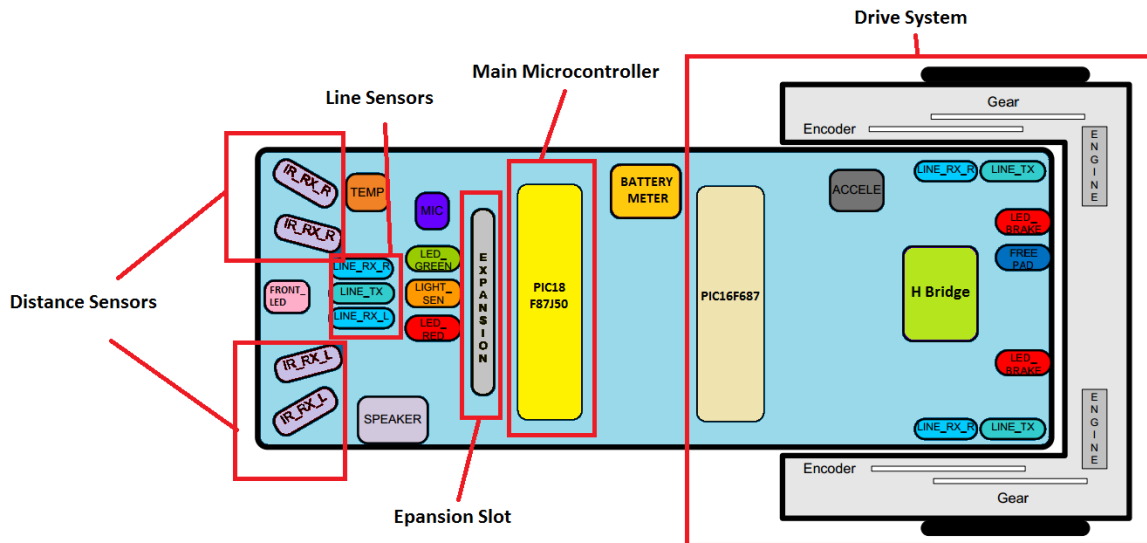


Figura III.61: Prezentare a roboților mOway

- dark → lumina este prea slabă pentru a se reflecta și senzorul înregistrează o tensiune ridicată.

Senzorul de detecție a obstacolelor utilizează o lumină infraroșie pentru a detecta obiectele din fața robotului mobil mOway. Acest senzor are două surse emițătoare (Kingbright KPA3010-F3C) și patru receptoare (PT100F0MP) amplasate pe ambele părți în fața robotului. Ieșirea receptoarelor este conectată la intrarea analogică a microcontrolerului, astfel încât să detecteze prezența oricărui obiect și să măsoare distanța față de acesta. Emițătorul de lumină generează un impuls de 70us, care permite receptoarelor să detecteze obstacolele utilizând un filtru și o fază de amplificare. Odată ce semnalul este procesat electronic, microcontrolerul îl poate măsura utilizând un convertor analog-digital (ADC) sau ca o intrare digitală. Distanța măsurată este cuprinsă între 3 cm și 11 cm, iar culoarea și luminozitatea suprafeței influențează precizia senzorului. Pentru a minimiza efectele perturbațiilor asupra distanței citite, a fost atașată o placă albă în spatele roboților așa cum se vede în Fig. III.60. Modulele de comunicații radio (RF) sunt bidirecționale, dar există o particularitate: comunicarea este semi-duplex. Potrivit Tanenbaum [291], un sistem semi-duplex asigură comunicarea în ambele direcții, cu toate acestea ele nu pot fi simultane (este permisă comunicarea într-o singură direcție la un moment dat). Caracteristicile principale ale modulului RF sunt: frecvența de lucru de 2,4 GHz, consum redus și viteză de transmisie între 1 și 2 Mbps.

III.3.2.2 Comunicați wireless

Una dintre sarcinile provocatoare a fost dezvoltarea unei strategii de utilizare a RFUsb ca dispozitiv MultiPoint-to-Point (MP2P) [292], deoarece configurația implicită pentru sistemul de comunicare mOway (folosind bibliotecile și transmițând fără nici un control de date) au dus la următoarele probleme: datele primite au fost prost structurate, multe pachete

au fost pierdute și nu a existat niciun parametru index care să informeze dacă un pachet era o transmisie sau o retransmisie.

Prima problemă care trebuia abordată a fost configurația pachetului de date. Datele importante care trebuiau colectate din sistem au fost: citirea senzorului de distanță (*ssr*), viteza (*spd*), distanța parcursă (*km*), starea bateriei (*batt*) și id-ul robotului care a transmis datele (*id*) care este monitorizată prin protocolul de comunicare, pentru fiecare robot mOway. Având în vedere faptul că roboții mOway ar putea trimite doar 8 octeți la un moment dat, structura pachetului a fost definită ca $[0, 0, batt, km_2, km_1, ssr, spd, id]$. Alte date care ar putea fi utile în structurarea vehiculelor în plutoane ar include: accelerația, direcția de rulare, poziția exactă, comportamentul viitor pentru utilizarea în strategiile de control distribuite [146] și alte informații importante privind siguranța, de exemplu accidente pe șosea, cozi la o intersecție.

Dezvoltarea părții PC a fost bazată pe o tehnică de interogare pentru a primi datele. Timpul necesar pentru a efectua un interogare completă pentru fiecare mOway se mărește cu numărul subiecților prezenți. Pentru a evita aceasta, a fost dezvoltată o tehnică pasivă (pentru PC) - activă (pentru mOway), în care modulul mOway trimite în mod continuu date către modulul RFUsb.

Timpul de eșantionare a fost prea rapid și instabil pentru ca RFUsb să proceseze datele și să aștepte să sosească următorul pachet de date. Acest fapt a dus, din nou, la pierderi de pachete și la diferite întârzieri, deoarece retransmisiile de la fiecare ieșire au avut loc în mod continuu. Ca atare, în loc de a schimba aplicația roboților, aplicația din partea PC-ului a fost reproiectată în care s-au utilizat buffere. În acest fel, timpul minim a fost atins de prelucrarea și stocarea efectivă a pachetului de date, evitându-se blocarea acestuia și, prin urmare, imposibilitatea de a primi mai multe date. Deși au fost înregistrate îmbunătățiri, acest lucru nu a fost suficient, deoarece timpul de prelevare de la fiecare mOway a fost inconsecvent și a provocat supraîncărcarea canalului de comunicații. Conform identificării efectuate, timpul minim de eșantionare ar putea ajunge până la 50 ms. Pentru a seta timpul de prelevare a datelor, algoritmul din roboții mOway a fost ușor modificat, astfel încât să accepte întreruperile temporizate și apoi procedura de trimitere a putut fi controlată.

După sincronizarea expedierii prin setarea unui impuls inițial atunci când sistemul era gata să înceapă, temporizatoarele din fiecare mOway ar putea arunca excepții în același timp. Pachetele au fost transmise, dar raportul dintre livrările de succes și expedierea totală a fost de aproximativ 25% până la 30%. Ca atare, acest raport a fost îmbunătățit prin stabilirea unui timp mai mare pentru întreruperile de trimitere, păstrând în același timp prelevarea de date la 50 ms. Deoarece numai 8 octeți puteau fi trimiși la un moment dat, datele au fost reduse și pachetul final a avut această configurație: $[km_2, km_1, ssr, spd, id, ssr, spd, id]$. Raportul de succes pe distanță este ilustrat în figurile următoare, în funcție de numărul de roboți mOway utilizați. Așa cum se poate observa din Fig. III.62, comportamentul de transmisie pentru un singur robot mOway a fost aproape perfect.

Când doi roboți mOway au fost puși împreună, rata de succes a scăzut imediat de la aproape 100% la aproximativ 60% și așa cum se poate observa din fig. III.63 distanța nu a influențat prea mult comunicațiile.

În cele din urmă, a fost setat un test cu patru roboți mOway și rata de transmisie este ilustrată în Fig. III.64. Rata de succes a scăzut de la 60% la 50%. Pierderea anormală a pachetelor

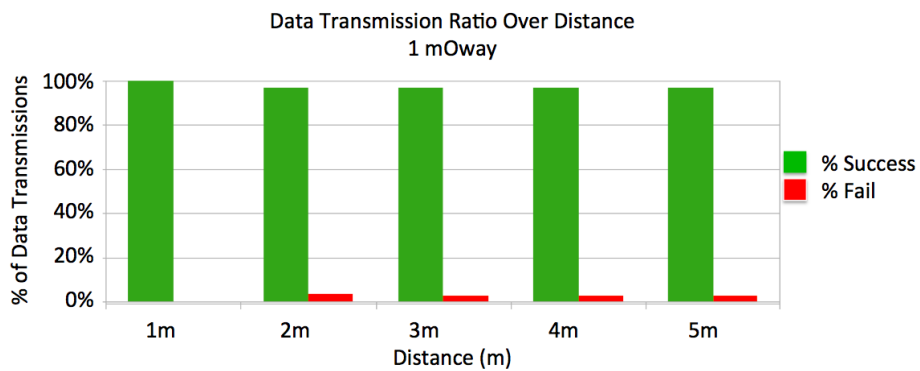


Figura III.62: Transmisia datelor cu succes a robotului mOway

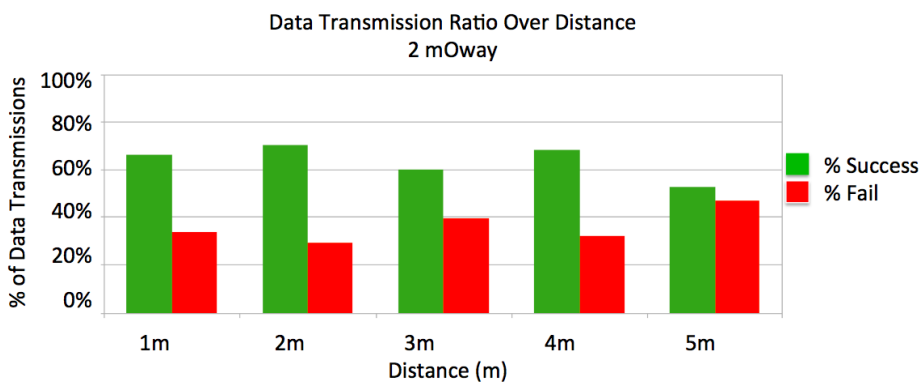


Figura III.63: Transmisia datelor cu succes a celor doi roboți mOway.

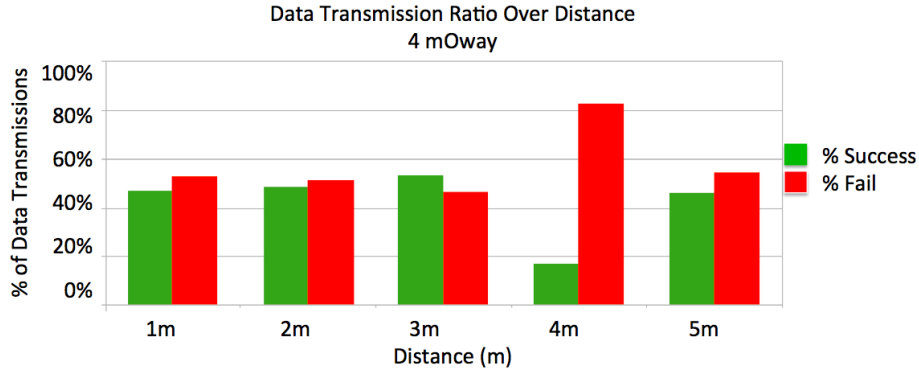


Figura III.64: Transmisia datelor cu succes a celor patru roboți mOway.

de date pentru testele efectuate la distanța de 4m prezentată în Fig. III.64 este corelată cu o problemă a transmițătorului unui robot.

III.3.2.3 Proiectarea regulatorului predictiv

Controlul predictiv bazat pe model (MPC) este o metodologie de control care utilizează un model al procesului pentru calcularea on-line a predicțiilor ieșirii procesului pe baza căreia optimizează acțiunile viitoare de control. De asemenea, constrângerile pot fi luate în considerare în această procedură de optimizare. Strategia MPC este ușor de înțeles și are un sens practic. În primul rând, un model de proces este utilizat pentru a predicta evoluția ieșirii procesului ca o funcție a acțiunilor viitoare de control. În al doilea rând, un indice de cost specificat este redus la minimum față de aceste acțiuni de control. Acest cost include în mod obișnuit erorile dintre ieșirile dorite și cele predictate ale procesului. Principiul MPC este descris în Fig. III.65. Referindu-ne la această figură, se urmărește următoarea strategie:

- la fiecare moment de timp k , ieșirea procesului $y(k+i)$ este predictată pe un orizont de timp $i = 1, \dots, h_p$; valorile predictate sunt indicate de $y(k+i|k)$ și valoarea h_p este denumită orizontul de predicție; predicția se face prin intermediul unui model al procesului și depinde de intrările și ieșirile din trecut, dar și de viitorul scenariu de control $\{u(k+i|k), i = 0, \dots, h_p - 1\}$;
- o traiectorie de referință $\{r(k+i|k), i = 1, \dots, h_p\}$, evoluând spre valoarea de referință w este definită pe orizontul de predicție, descriind modul în care ieșirea procesului tinde de la valoarea sa curentă $y(k)$ la valoarea de referință w ;
- vectorul de control $\{u(k+i|k), i = 0, \dots, h_p - 1\}$ este calculat pentru a minimiza o funcție de cost specificată, de exemplu, cea mai simplă funcție de cost, unde efortul de reglare nu este luat în considerare, este definită astfel:

$$\sum_{i=1}^{h_p} (r(k+i|k) - y(k+i|k))^2 \xrightarrow{\text{Minimize}} u(k+i|k). \quad (\text{III.35})$$

- primul element $u(k|k)$ al comenzii optimale este trimis către proces, toate celelalte ele-

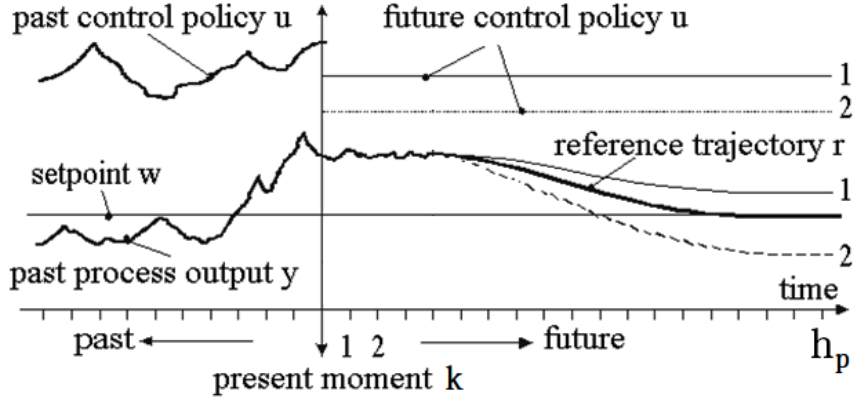


Figura III.65: Principiul MPC

mente ale vectorului de control fiind înlăturate.

Ca atare, la următoarea perioadă de eșantionare se repetă această optimizare, luând în considerare noile informații de măsurare. Aceasta introduce, de fapt, componenta de feedback din întreaga strategie, rezultând o configurație în buclă închisă.

Considerăm modelul discret al robotului mOway (III.23)

$$\mathbf{z}(k+1) = \mathbf{A}_d \mathbf{z}(k) + \mathbf{B}_d u(k), \quad k \in \mathbb{Z}_+, \quad (\text{III.36})$$

unde $\mathbf{z}(k) \in \mathbb{R}^2$ este vectorul stărilor, $u(k) \in \mathbb{R}$ este semnalul de control (referința pentru viteza unghiulară a roților robotului) la momentul discret de timp k și \mathbf{A}_d și \mathbf{B}_d sunt matricile discrete ale sistemului.

Pentru roboții mobili, modelul (III.36) a fost obținut prin discretizarea modelului (III.23) cu o perioadă de eșantionare $T_s = 5$ ms și este definit de următoarele matrici ale sistemului

$$\mathbf{A}_d = \begin{pmatrix} 1 & 0.0287 \\ 0 & 0.2901 \end{pmatrix}, \quad \mathbf{B}_d = \begin{pmatrix} 0.0047 \\ 0.1570 \end{pmatrix} \quad (\text{III.37})$$

unde $\mathbf{z}(k) = [x(k) \quad v(k)]$, unde $x(k)$ este poziția absolută și $v(k)$ este viteza robotului la momentul de timp k .

Stările prezise pot fi scrise în forma următoare:

$$\hat{\mathbf{Z}}(k) = \mathbf{M} \mathbf{z}(k) + \mathbf{C} \mathbf{U}(k), \quad (\text{III.38})$$

unde

$$\mathbf{U}(k) := \begin{pmatrix} u(k|k) \\ u(k+1|k) \\ \vdots \\ u(k+h_p-1|k) \end{pmatrix}, \quad \hat{\mathbf{Z}}(k) := \begin{pmatrix} \hat{\mathbf{z}}(k+1|k) \\ \hat{\mathbf{z}}(k+2|k) \\ \vdots \\ \hat{\mathbf{z}}(k+h_p|k) \end{pmatrix}, \quad (\text{III.39})$$

$u(k+i|k)$, $i = 0, \dots, h_p - 1$, este secvența de control viitoare cu orizontul de predicție h_p , $\hat{\mathbf{z}}(k+i|k)$ este valoarea prezisă.

$$\begin{aligned} \hat{\mathbf{z}}(k+i+1|k) &= \\ &= \mathbf{A}_d \hat{\mathbf{z}}(k+1|k) + \mathbf{B}_d u(k+i|k), \quad i = 0, 1, 2, \dots \end{aligned} \quad (\text{III.40})$$

avand condiția inițială definită astfel

$$\hat{\mathbf{z}}(k|k) = \mathbf{z}(k), \quad (\text{III.41})$$

$$\mathbf{M} = \begin{bmatrix} \mathbf{A}_d \\ \mathbf{A}_d^2 \\ \vdots \\ \mathbf{A}_d^{h_p} \end{bmatrix}, \mathbf{C} = \begin{bmatrix} \mathbf{B}_d & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{A}_d \mathbf{B}_d & \mathbf{B}_d & \cdots & \mathbf{0} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{A}_d^{h_p-1} \mathbf{B}_d & \mathbf{A}_d^{h_p-2} \mathbf{B}_d & \cdots & \mathbf{B}_d \end{bmatrix}. \quad (\text{III.42})$$

Legea de control predictiv este calculată astfel

$$\mathbf{U}^*(k) = \arg \min_{\mathbf{U}} J(k) \quad (\text{III.43})$$

și se bazează pe minimizarea unei funcții de cost

$$J(k) = \hat{\mathbf{z}}(k + h_p|k)^T \bar{\mathbf{Q}} \hat{\mathbf{z}}(k + h_p|k) + \sum_{i=0}^{h_p-1} (\hat{\mathbf{z}}(k + i|k)^T \mathbf{Q} \hat{\mathbf{z}}(k + i|k) + u(k + i|k)^T \mathbf{R} u(k + i|k)), \quad (\text{III.44})$$

care poate fi definită corelat cu secvența de comandă viitoare \mathbf{U} și starea viitoare predictată $\hat{\mathbf{Z}}$ având forma matriceală următoare

$$J(k) = \mathbf{U}^T(k) \mathbf{H} \mathbf{U}(k) + 2 \hat{\mathbf{Z}}^T(k) \mathbf{F}^T \hat{\mathbf{Z}}(k) + \hat{\mathbf{Z}}^T(k) \mathbf{G} \hat{\mathbf{Z}}(k), \quad (\text{III.45})$$

unde

$$\begin{aligned} \mathbf{H} &= \mathbf{C}^T \tilde{\mathbf{Q}} \mathbf{C} + \tilde{\mathbf{R}} \\ \mathbf{F} &= \mathbf{C}^T \tilde{\mathbf{Q}} \mathbf{M} \\ \mathbf{G} &= \mathbf{M}^T \tilde{\mathbf{Q}} \mathbf{M} + \mathbf{Q}, \end{aligned} \quad (\text{III.46})$$

cu

$$\tilde{\mathbf{Q}} = \begin{bmatrix} \mathbf{Q} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \ddots & & \vdots \\ \vdots & & \mathbf{Q} & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \bar{\mathbf{Q}} \end{bmatrix} \text{ and } \tilde{\mathbf{R}} = \begin{bmatrix} \mathbf{R} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \ddots & & \vdots \\ \vdots & & \mathbf{R} & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{R} \end{bmatrix} \quad (\text{III.47})$$

unde \mathbf{Q} , \mathbf{R} și $\bar{\mathbf{Q}}$ sunt matrici pozitiv definite (matricea \mathbf{Q} poate fi pozitiv semi-definită) cu dimensiunile corespunzătoare.

În cazul în care nu sunt definite restricții, soluția care minimizează funcția de cost (III.43) poate fi calculată astfel

$$\mathbf{U}^*(k) = -\mathbf{H}^{-1} \mathbf{F} \mathbf{z}(k) \quad (\text{III.48})$$

și în conformitate cu principiul orizontului alunecător, către proces va fi trimisă doar valoarea \mathbf{U}

$$u(k) = u^*(k|k) = \mathbf{K}_{h_p} \mathbf{z}(k) \quad (\text{III.49})$$

unde $\mathbf{K}_{h_p} = -[\mathbf{I}_{h_p} \quad \mathbf{0} \quad \cdots \quad \mathbf{0}] \mathbf{H}^{-1} \mathbf{F}$.

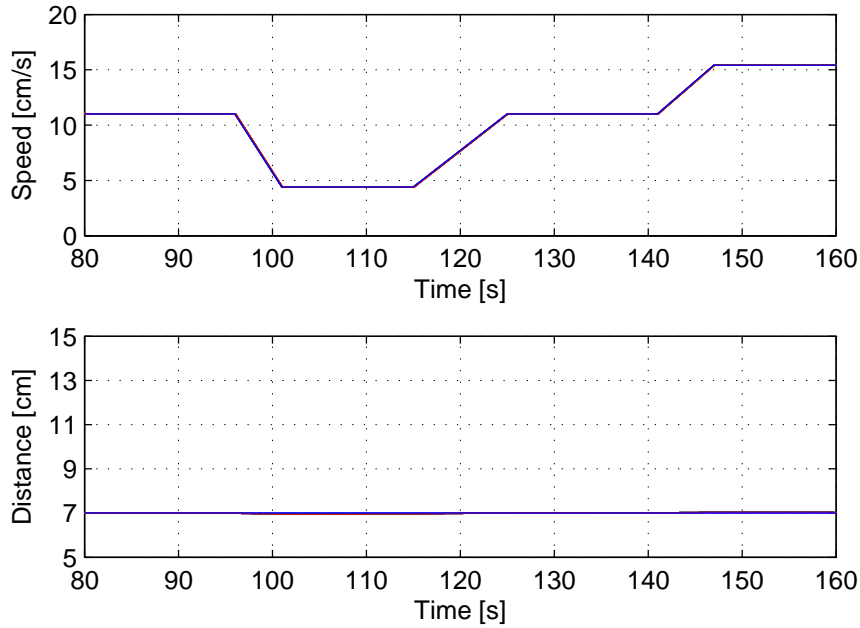


Figura III.66: Simulare: vitezele și distanțele robotilor mobili

III.3.2.4 Rezultate de simulare și experimentale

Regulatorul a fost conceput pentru a minimiza eroarea $e(k) = d(k) - r(k)$, unde $d(k)$ este distanța dintre robotul curent și robotul din față lui i , $r(k)$ este distanța de referință pentru $k \in \mathbb{Z}_+$. În cazul nostru, $r(k) = r$ a fost considerat egal cu 7 cm, pentru toate $k \in \mathbb{Z}_+$. De reținut este că o referință pentru prima stare, adică poziția robotului mobil, poate fi calculată ca $x(k) = x(k-1) + e(k)$. Distanța $d(k)$ a fost măsurată la fiecare perioadă de eșantionare folosind senzorul de obstacol, iar a doua stare (viteza robotului) nu a fost măsurată, ci calculată ca derivata primei stări (poziția robotului).

În Fig. III.66 sunt ilustrate rezultatele simulării obținute cu regulatorul MPC pe stare, pentru care parametrii de proiectare au fost aleși pentru a obține un compromis între suprareglare și rapiditate, cu un orizont de predicție setat la $h_p = 5$. Parametrii regulatorului MPC au fost aleși după cum urmează: $\bar{\mathbf{Q}} = 2\mathbf{I}_2$; $\mathbf{Q} = 1.5\mathbf{I}_2$; $\mathbf{R} = 1$; nu au fost incluse constrângeri în problema de optimizare. Comunicarea V2I a fost utilizată numai pentru colectarea de date pentru analiza comportamentului offline.

Rezultatele experimentale obținute cu același regulator MPC pe stare sunt reprezentate în Fig. III.67, în care se poate observa că eroarea de distanță este apropiată de 0. De reținut este faptul că răspunsurile oscilatorii sunt de asemenea datorate efectelor zgomotului pe senzorul de distanță (la citire). Se poate observa că vitezele urmărilor sunt mai mari decât viteza liderului, deoarece următorii trebuie să călătorească mai mult pentru a menține o distanță de referință predefinită față de robotul mobil din față, ceea ce înseamnă că se atinge o viteză medie mai mare (a se vedea subsecțiunea III.3.2.5 pentru detalii suplimentare).

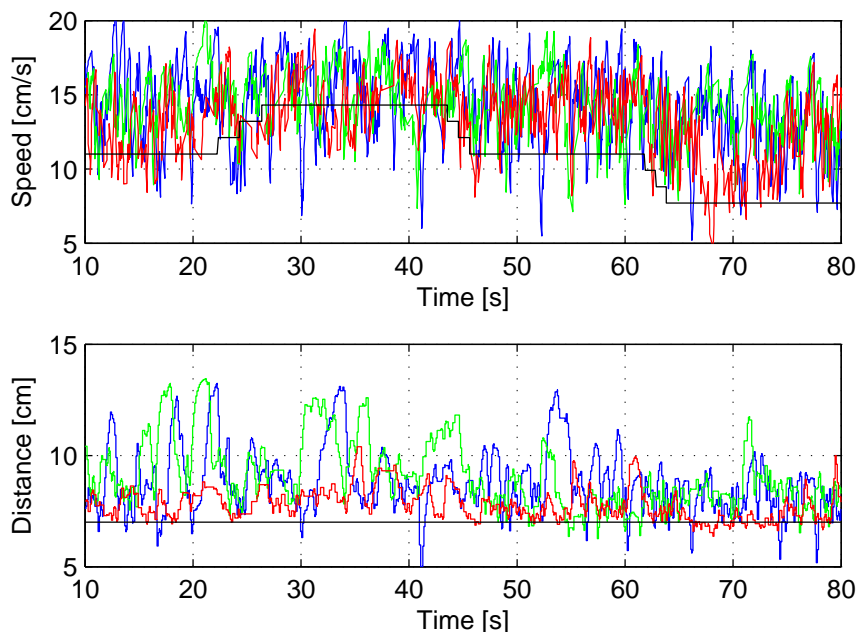


Figura III.67: Experiment: vitezele și distanțele roboților mobili

III.3.2.5 Remarci asupra rezultatelor experimentale

În Fig. III.67 se poate observa că viteza medie a roboților urmăritori este continuu mai mare decât viteza robotului lider. Acest lucru se datorează mai multor circumstanțe în ceea ce privește configurația experimentală după cum urmează:

- în experimentele efectuate toate roboții mobili urmează o linie marcată pe podea;
- roboții mobili nu rămân perfect pe linie în timp ce rulează, dar dacă apare o abatere de la traiectoria impusă, această abatere este considerată ca o perturbație [276] care este rejectată prin modificarea vitezei unghiulare a motorului din stânga printr-un avans *gain*; totuși, aceasta înseamnă că trebuie să călătorească mai mult pentru a menține o distanță de referință predefinită față de robotul mobil din față ceea ce înseamnă că se atinge o viteză medie mai mare;
- mișcarea în jurul liniei pe care ar trebui să o urmeze roboții influențează de asemenea distanța măsurată de fiecare robot mobil (dacă liderul este direct pe linie și urmăritorul este orientat puțin spre stânga / dreapta, distanța măsurată de acesta este practic mai mare decât distanța reală care corespunde de fapt unei viteze medii mai mari);
- când liderul ajunge într-o curbă, urmăritorul nu reușește să detecteze exact distanța (de obicei, distanța măsurată este mai mare), ceea ce determină creșterea rapidă a vitezei pentru a reduce eroarea artificială.

III.3.2.6 Remarci asupra comunicației

Pe baza datelor colectate de roboții mobili de la mOway, putem concluziona că:

- sistemul de comunicare V2I rulează fără probleme semnificative;
- cele 4 roboți mobili implicați în determinarea rezultatei experimentale pot păstra o

distanță de siguranță pentru a conduce într-un scenariu de tip pluton;

- din cauza limitărilor hardware ale sistemului de comunicație, datele colectate au fost strict legate de dinamica roboților mobili; acestea ar putea fi actualizate cu diferiți senzori pentru a furniza informații aprofundate despre mediul înconjurător;
- cercetarea ar putea beneficia de includerea mai multor roboți mobili în pluton și de îmbunătățirea stabilității sistemului de comunicație;
- roboții ar putea comunica direct între ei folosind comunicații V2V pentru fixarea și optimizarea procesului de luare a deciziilor.

III.4 (A3.3) Analiza performanțelor în timp real a algoritmilor predictivi bazați pe agenți și compararea cu soluțiile existente pentru plutoanele de vehicule

III.4.1 Descrierea performanțelor în timp real a algoritmului de tip PID

Odată plutonul de vehicule realizat și testat, au fost realizate primele experimente privind reglarea distanței dintre vehiculele din pluton în diverse condiții. Algoritmul de control implementat a fost de tip PID. Pentru început experimentele au fost realizate cu plutoane formate din 2 vehicule, după cum este ilustrat în Fig. III.68. Algoritmii de control de tip PID nu au cerințe ridicate privind puterea de calcul, astfel încât ambii algoritmi (pentru controlul traiectoriei și pentru controlul longitudinal) au fost implementați în cadrul aplicației încorporate bazate pe sistemul de operare de timp real FreeRTOS ce rulează pe placa principală.

Pentru plutonul compus din două vehicule referința a fost setată ca fiind 120mm. Distanța față de mașina din față în cadrul unui pluton cu 2 vehicule este ilustrată în Fig. III.69. De asemenea, comportamentul regulatorului longitudinal atunci când mașina se alătură plutonului poate fi observat în Fig. III.70. Se poate observa că atâta timp cât senzorul de distanță cu infraroșu nu detectează mașina din față ca fiind la o distanță între 100mm și 300mm, distanța înregistrată este de 310mm. Imediat cum mașina din față este detectată, controllerul de distanță modifică viteza mașinii pentru a obține distanța de referință de 120mm.

În continuare vor fi prezentate rezultatele obținute cu cele trei scheme de control cu regulatoare PID, CACC-GPC și DMPC. Pentru toate experimentele realizate s-a considerat o referință de distanță impusă de 15cm = 0.15m.

Fig. III.71 prezintă performanța algoritmului PID pentru un pluton cu 6 vehicule. În acest caz placa STM32F3Discovery nu a fost utilizată. În Fig. III.71 sunt ilustrate distanțele dintre fiecare urmăritor și liderul aparent (de exemplu, primul urmăritor este lider aparent pentru urmăritorul al doilea) așa cum au fost obținute experimental cu structuri de reglare folosind regulatoare PID.

Se poate observa că referința impusă este urmarită, însă cu oscilații notabile. Suprareglările sunt amplificate spre capătul plutonului, ceea ce denotă instabilitatea plutonului de vehicule. Se mai poate observa de asemenea că timpul de răspuns este aproximativ egal cu 4 secunde pentru primul urmăritor și crește spre capătul plutonului.

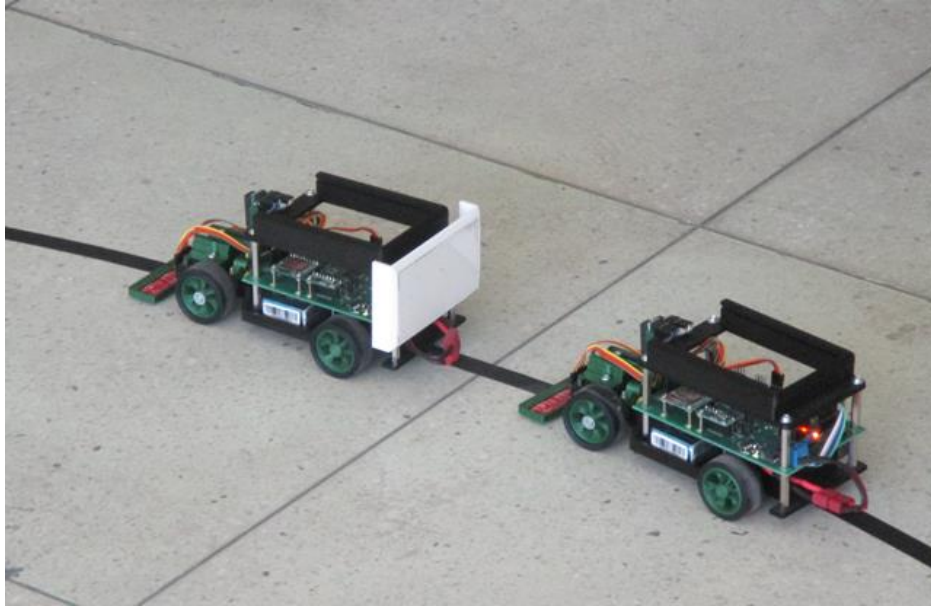


Figura III.68: Experiment cu pluton format din 2 vehicule.

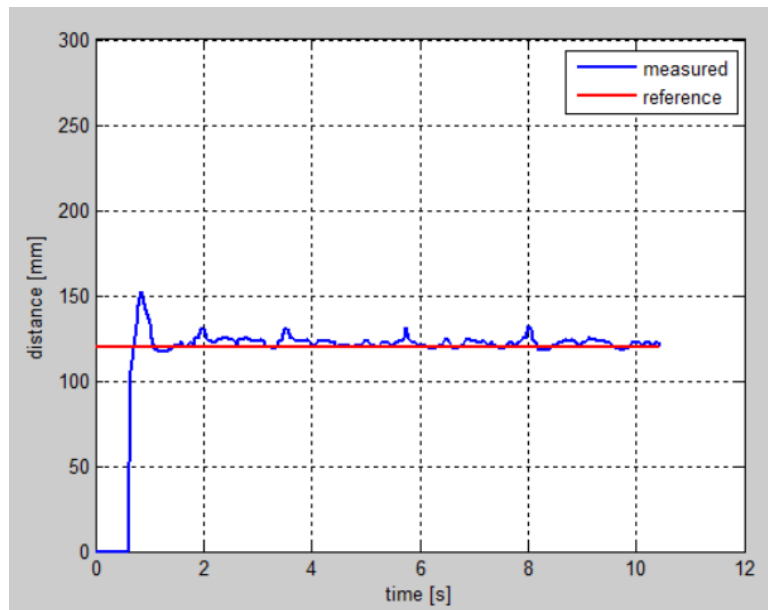


Figura III.69: Distanța față de mașina din față într-un pluton.

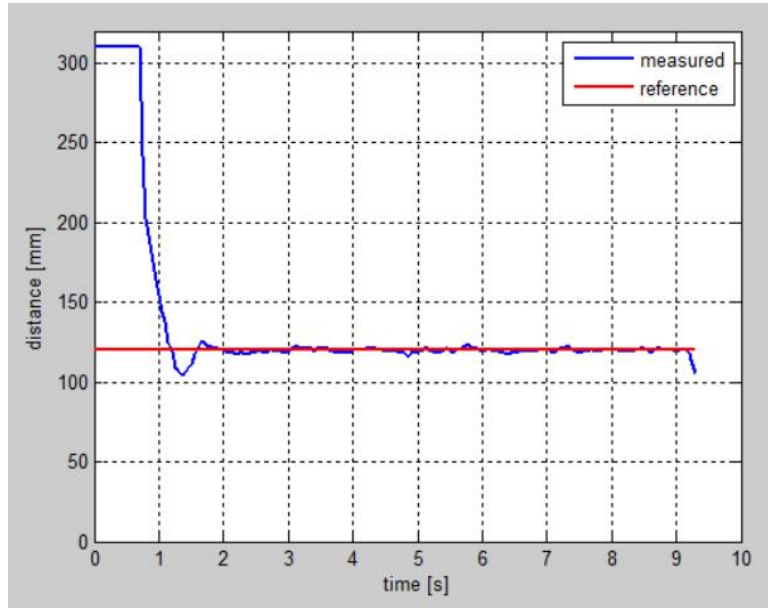


Figura III.70: Distanța față de mașina din față la alăturarea la un pluton.

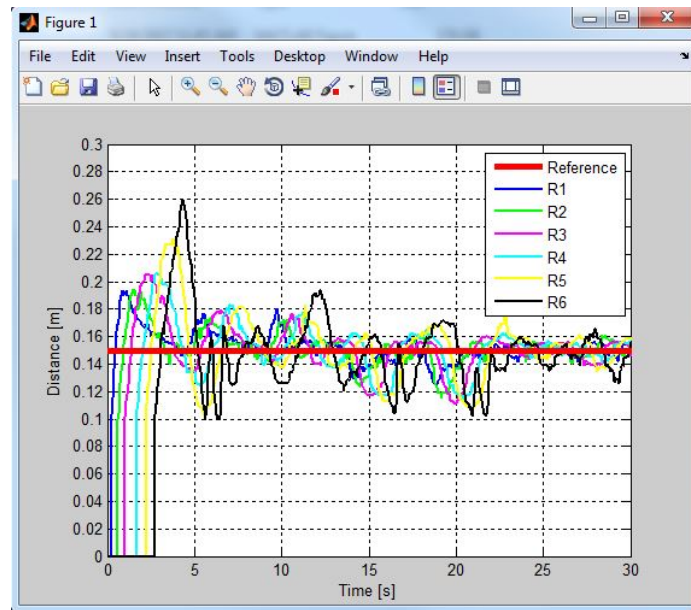


Figura III.71: Experiment cu pluton format din 6 vehicule și algoritm de control PID.

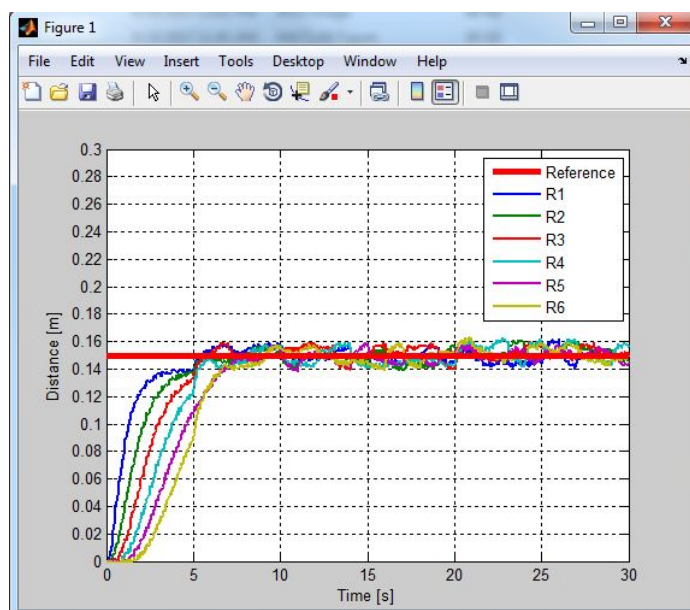


Figura III.72: Experiment cu pluton format din 6 vehicule și algoritm de control GPC.

III.4.2 Descrierea performanțelor în timp real a algoritmului de tip CACC-GPC

În Fig. III.72 sunt ilustrate distanțele dintre fiecare urmăritor și liderul așa cum au fost obținute experimental cu structuri de reglare folosind regulatoare CACC-GPC.

Ca și în cazul structurilor de control cu regulatoare PID, și în acest caz, se poate observa că referința impusă este urmarită, însă cu oscilații mult mai mici. Oscilațiile nu mai sunt amplificate spre capătul plutonului, ceea ce demonstrează că plutonul de vehicule este stabil. Se mai poate observa de asemenea că timpul de răspuns este mai mare față de cel obținut cu regulatoare PID, fiind aproximativ egal cu 5 secunde pentru toți urmăritorii.

III.4.3 Descrierea performanțelor în timp real a algoritmului de tip DMPC

În Fig. III.73 sunt ilustrate distanțele dintre fiecare urmăritor și liderul aparent așa cum au fost obținute experimental cu structuri de reglare folosind regulatoare DMPC.

Ca și în cazul structurilor de control cu regulatoare PID și CACC-GPC, și în acest caz, se poate observa că referința impusă este urmarită, de această dată cu oscilații foarte mici. Oscilațiile nu sunt amplificate spre capătul plutonului, ceea ce demonstrează că plutonul de vehicule este stabil, ca și în cazul regulatoarelor de tipul CACC-GPC. Se mai poate observa de asemenea că timpul de răspuns este mult mai mic decât cele obținute cu regulatoare PID și CACC-GPC, fiind aproximativ egal cu 1.3 secunde pentru toți urmăritorii.

În concluzie, rezultatele experimentale obținute cu cele 3 tipuri de regulatoare (PID, CACC-GPC și DMPC) evidențiază faptul că regulatorul PID ar putea constitui o soluție pentru controlul vehiculelor din plutoanele de dimensiuni reduse (3-4 vehicule per pluton). Cu toate că algoritmi de control predictiv distribuți bazați pe agenți dezvoltați în cadrul proiectului presupun o structură mai complexă (transmiterea de informații între vehicule viteză și mărimi de comandă), performanțele obținute sunt mult mai bune în comparație cu cele

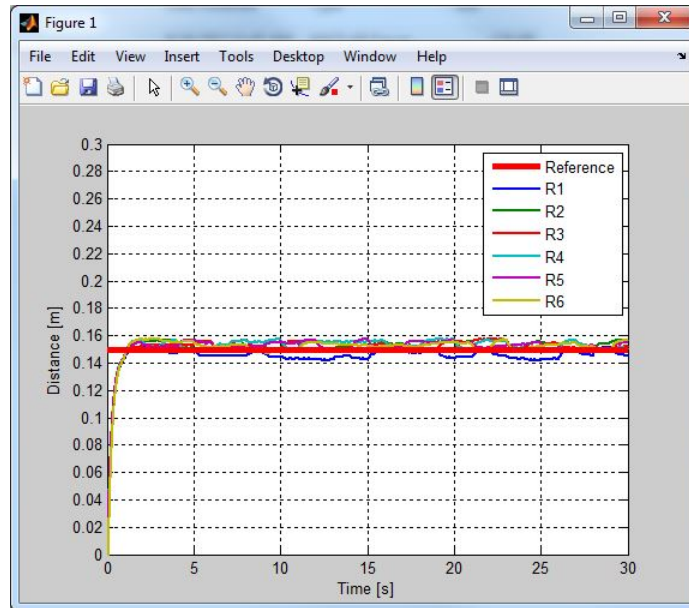


Figura III.73: Experiment cu pluton format din 6 vehicule și algoritm de control DMPC.

pbținute cu reglatoare PID, ceea ce justifică o astfel de abordare pentru plutoane de vehicule în care să fie asigurată siguranța și stabilitatea vehiculelor și a plutonului ca întreg.

Bibliografie

- [1] Directorate-General for Mobility and Transport — European Commission, Brussels, “Smarter transport systems mean safer roads,” *Road Safety Newsletter*, 2012.
- [2] European Commission, *EU transport in figures*. Luxembourg: Publications Office of the European Union, 2015.
- [3] Y. Sugiyama, M. Fukui, M. Kikuchi, K. Hasebe, A. Nakayama, K. Nishinari, S. Tadaki, and S. Yukawa, “Traffic jams without bottlenecks - experimental evidence for the physical mechanism of the formation of a jam,” *New Journal of Physics*, vol. 10, pp. 1–7, 2008.
- [4] Y. Zhao, P. Minero, and V. Gupta, “On disturbance propagation in vehicle platoon control systems,” in *American Control Conference*, Montreal, Canada, 2012.
- [5] G. Naus, R. Vugts, J. Ploeng, M. van de Molengraft, and M. Steinbuch, “String-stable cacc design and experimental validation: A frequency-domain approach,” *IEEE Transactions on Vehicular Technology*, vol. 59, pp. 4268–4279, 2010.
- [6] Y.-B. Zhao, G.-P. Liu, and D. Rees, “Packet-based deadband control for Internet-based networked control systems,” *IEEE Transactions on Control Systems Technology*, vol. 18, pp. 1057–1067, 2010.
- [7] S. Oncu, N. van de Wouw, W. Heemels, and H. Nijmeijer, “String stability of interconnected vehicles under communication constraints,” in *51st IEEE Conference on Decision and Control*, Maui, Hawaii, USA, 2012.
- [8] D. Jia, K. Lu, and J. Wang, “A disturbance-adaptive design for VANET-enabled vehicle platoon,” *IEEE Transactions on Vehicular Technology*, vol. 63, pp. 527–539, 2014.
- [9] Z. Gacovski and S. Deskovski, “Different control algorithms for a platoon of autonomous vehicles,” *International Journal of Robotics and Automation*, vol. 3, pp. 151–160, 2014.
- [10] D. Swaroop and J. Hedrick, “String stability of interconnected systems,” *IEEE Transactions on Automatic Control*, vol. 41, pp. 349–357, 1996.
- [11] A. Vahidi and A. Eskandarian, “Research advances in intelligent collision avoidance and adaptive cruise control,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 4, pp. 143–153, 2003.
- [12] A. Gray, Y. Gao, T. Lin, J. Hedrick, and F. Borrelli, “Stochastic predictive control for semi-autonomous vehicles with an uncertain driver model,” in *16th IEEE Annual International Conference on Intelligent Transportation Systems*, The Hague, The Netherlands, 2013, pp. 2329–2334.
- [13] C. F. Caruntu, C. Copot, C. Lazar, and R. De Keyser, “Longitudinal control of vehicle platoons for stop-and-go waves mitigation,” in *18th International Conference on System Theory, Control and Computing*, Sinaia, Romania, 2014, pp. 670–675.

- [14] S. Lefevre, A. Carvalho, and F. Borelli, “Autonomous car following: A learning-based approach,” in *IEEE Intelligent Vehicles Symposium*, Seoul, Korea, 2015, pp. 920–926.
- [15] G. Schildbach and F. Borelli, “Scenario model predictive control for lane change assistance on highways,” in *IEEE Intelligent Vehicles Symposium*, Seoul, Korea, 2015, pp. 611–616.
- [16] S. Yoon, H. Park, J. Ryu, and M. Lee, “Lateral control of obstacle avoidance for an autonomous vehicle with laser scanner,” in *IEEE International Symposium on Industrial Electronics*, Cambridge, UK, 2008, pp. 1015–1019.
- [17] G. Klancar, D. Matko, and S. Blazic, “Wheeled mobile robots control in a linear platoon,” *Journal of Intelligent and Robotic Systems*, vol. 54, pp. 709–731, 2009.
- [18] K.-Y. Liang, A. Alam, and A. Gattami, “The impact of heterogeneity and order in heavy duty vehicle platooning networks,” in *IEEE Vehicular Networking Conference*, Amsterdam, The Netherlands, 2011.
- [19] S. Cristescu, C. Ionescu, B. Wyns, R. De Keyser, and I. Nascu, “Leader-follower string formation using cascade control for mobile robots,” in *20th Mediterranean Conference on Control and Automation*, Barcelona, Spain, 2012.
- [20] A. Alam, J. Martensson, and K. Johansson, “Look-ahead cruise control for heavy duty vehicle platooning,” in *16th International IEEE Annual Conference on Intelligent Transportation Systems*, The Hague, The Netherlands, 2013.
- [21] A. Scheuer, O. Simonin, and F. Charpillat, “Safe longitudinal platoons of vehicles without communication,” in *IEEE International Conference on Robotics and Automation*, Kobe, Japan, 2009.
- [22] G. Klancar, D. Matko, and S. Blazic, “A control strategy for platoons of differential drive wheeled mobile robot,” *Robotics and Autonomous Systems*, vol. 59, pp. 57–64, 2011.
- [23] V. Turri, B. Besselink, J. Martensson, and K. Johansson, “Fuel-efficient heavy-duty vehicle platooning by look-ahead control,” in *53rd IEEE Conference on Decision and Control*, Los Angeles, USA, 2014.
- [24] M. Segata, B. Bloessl, S. Joerer, F. Dressler, and R. L. Cigno, “Supporting platooning maneuvers through IVC: an initial protocol analysis for the join maneuver,” in *11th IEEE/IFIP Conference on Wireless on Demand Network Systems and Services*, Obergurgl, Austria, 2014.
- [25] S.-Y. Han, Y.-H. Chen, L. Wang, and A. Abraham, “Decentralized longitudinal tracking control for cooperative adaptive cruise control systems in a platoon,” in *IEEE Conference on Systems, Man and Cybernetics*, Manchester, UK, 2013, pp. 2013–2018.

- [26] B. Nemeth and P. Gaspar, "LPV-based control design of vehicle considering road inclinations," in *18th IFAC World Congress*, Milano, Italy, 2011, pp. 3837–3842.
- [27] P. Fernandes and U. Nunes, "Platooning with IVC-enabled autonomous vehicles: Strategies to mitigate communication delays, improve safety and traffic flow," *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, pp. 91–106, 2012.
- [28] M. Lammert, A. Duran, J. Diez, K. Burton, and A. Nicholson, "Effect of platooning on fuel consumption of class 8 vehicles over a range of speeds, following distances and mass," *SAE International Journal on Commercial Vehicles*, vol. 7, 2014.
- [29] H. Zhou, R. Saigal, F. Dion, and L. Yang, "Vehicle platoon control in high-latency wireless communications environment: Model predictive control method," *Journal of the Transportation Research Board*, vol. 2324, pp. 81–90, 2012.
- [30] A. Maxim, C. Ionescu, C. Caruntu, C. Lazar, and R. De Keyser, "Reference tracking using a non-cooperative distributed model predictive control algorithm," in *11th IFAC Symposium on Dynamics and Control of Process Systems, including Biosystems*, Trondheim, Norway, 2016, pp. 1079–1084.
- [31] A. Bose and P. A. Ioannou, "Analysis of traffic flow with mixed manual and semiautomated vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 4, pp. 173–188, 2003.
- [32] A. Monot, N. Navet, B. Bavoux, and F. Simonot-Lion, "Multisource software on multicore automotive ECUs combining runnable sequencing with task scheduling," *IEEE Transactions on Industrial Electronics*, vol. 59, pp. 3934–3942, 2012.
- [33] Y. Tipsuwan and M. Y. Chow, "Control methodologies in networked control systems," *Control Engineering Practice*, vol. 11, pp. 1099–1111, 2003.
- [34] C. F. Caruntu, M. Lazar, R. H. Gielen, P. P. J. van den Bosch, and S. Di Cairano, "Lyapunov based predictive control of vehicle drivetrains over CAN," *Control Engineering Practice*, vol. 21, pp. 1884–1898, 2013.
- [35] C. F. Caruntu and C. Lazar, "Network delay predictive compensation based on time-delay modeling as disturbance," *International Journal of Control*, vol. 87, pp. 2012–2026, 2014.
- [36] M. B. G. Cloosterman, L. Hetel, N. van de Wouw, W. P. M. H. Heemels, J. Daafouz, and H. Nijmeijer, "Controller synthesis for networked control systems," *Automatica*, vol. 46, pp. 1584–1594, 2010.
- [37] X. Liu, Y. Xia, M. S. Mahmoud, and Z. Deng, "Modeling and stabilization of MIMO networked control systems with network constraints," *International Journal of Innovative Computing, Information and Control*, vol. 6, pp. 4409–4419, 2010.
- [38] P. Zometa, M. Kogel, T. Faulwasser, and R. Findeisen, "Implementation aspects of model predictive control for embedded systems," in *American Control Conference*, Montreal, Canada, 2012, pp. 1205–1210.

- [39] J. Jerez, G. Constantinides, E. Kerrigan, and K.-V. Ling, “Parallel MPC for real-time FPGA-based implementation,” in *18th IFAC World Congress*, Milano, Italy, 2011, pp. 1338–1343.
- [40] J. Anderson, N. Kalra, K. Stanley, P. Sorensen, C. Samaras, and O. Oluwatola, *Autonomous Vehicle Technology. A Guide for Policymakers*, RAND Corporation, 2014.
- [41] A. Gray, Y. Gao, J. Hedrick, and F. Borrelli, “Robust predictive control for semi-autonomous vehicles with an uncertain driver model,” in *IEEE Intelligent Vehicles Symposium*, Gold Coast, Australia, 2013, pp. 208–213.
- [42] U. Ozguner, T. Acarman, and K. Redmill, “Autonomous ground vehicles,” *chapters 1-4, Artech House*, 2012.
- [43] T. Keviczky, P. Falcone, F. Borrelli, J. Asgari, and D. Hrovat, “Predictive control approach to autonomous vehicle steering,” in *American Control Conference*, Minneapolis, US, 2006, pp. 4760–4675.
- [44] M. El-Zaher, B. Dafflon, F. Gechter, and J.-M. Contet, “Vehicle platoon control with multi-configuration ability,” in *International Conference on Computational Science*, 2012.
- [45] M. Ammozadeh, H. Deng, C.-N. Chuah, H. M. Zhang, and D. Ghosal, “Platoon management with cooperative adaptive cruise control enabled by VANET,” *Vehicular Communications*, vol. 2, pp. 110–123, 2015.
- [46] R. Meraihi, S.-M. Senouci, D.-E. Meddour, and M. Jerbi, “Vehicle-to-vehicle communications: Applications and perspectives,” in *Wireless Ad Hoc and Sensor Networks*, H. Labiod, Ed. Wiley, 2008, ch. 12, pp. 315–339.
- [47] M. Tomizuka, P. Hingwe, J.-Y. Wang, and M. Tai, *Intelligent control of road vehicles for automated driving: path architecture for automated highway systems and lateral guidance*. Oxford: Eolss Publishers Co Ltd, 2002, vol. Control systems, Robotics, and Automation, UNESCO Encyclopaedia of Life Support Systems <http://www.eolss.net>, Article contribution 6.43.36.02, 14 pages.
- [48] C. M. Kang, S. H. Lee, and C. C. Chung, “Comparative evaluation of dynamic and kinematic vehicle models,” in *53rd IEEE Conference on Decision and Control*, Los Angeles, CA, USA, 2014, pp. 648–653.
- [49] J. Kong, M. Pfeiffer, G. Schildbach, and F. Borrelli, “Kinematic and dynamic vehicle models for autonomous driving control design,” in *IEEE Intelligent Vehicles Symposium*, Seoul, Korea, 2015, pp. 1094–1099.
- [50] A. Gray, M. Ali, Y. Gao, J. Hedrick, and F. Borrelli, “A unified approach to threat assessment and control for automotive active safety,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 14, pp. 1490–1499, 2013.

- [51] H. B. Pacejka, E. Bakker, and L. Nyborg, “Tyre modelling for use in vehicle dynamics studies,” in *Society of Automotive Engineers*, 1987.
- [52] H. Pacejka, *Tire and Vehicle Dynamics*. Elsevier, 2012.
- [53] R. Rajamani, *Vehicle Dynamics and Control*. Springer, 2012.
- [54] L. Grune and J. Pannek, *Nonlinear Model Predictive Control*. Springer, 2011.
- [55] U. Kiencke and L. Nielsen, *Automotive control systems: for engine, driveline, and vehicle*. Springer Verlag, 2005, vol. 290.
- [56] M. Grotjahn, L. Quernheim, and S. Zemke, “Modelling and identification of car driveline dynamics for anti-jerk controller design,” in *IEEE International Conference on Mechatronics*, Budapest, Hungary, 2006, pp. 131–136.
- [57] C. F. Caruntu, A. E. Balau, M. Lazar, P. P. J. van den Bosch, and S. Di Cairano, “Driveline oscillations damping: A tractable predictive control solution based on a piecewise affine model,” *Nonlinear Analysis: Hybrid Systems*, vol. 19, pp. 168–185, 2016.
- [58] C. F. Caruntu, C. Copot, C. Lazar, and R. De Keyser, “Vehicle platooning for stop-and-go waves mitigation with an application to moway mobile robots,” *submitted to European Journal of Control journal*, 2018.
- [59] C. Copot, C. F. Caruntu, and R. De Keyser, “Advanced control technique to mitigate the stop-and-go waves on a highway traffic with different vehicles,” in *21st International Conference on Methods and Models in Automation and Robotics*, Miedzyzdroje, Poland, 2016, pp. 948–953.
- [60] R. Kianfar, P. Falcone, and J. Fredriksson, “A control matching-based predictive approach to string stable vehicle platooning,” in *19th IFAC World Congress*, Cape Town, South Africa, 2014, pp. 10 700–10 705.
- [61] A. Ali, G. Garcia, and P. Martinet, “Safe highways platooning with minimized inter-vehicle distances of the time headway policy,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems, 5th Workshop on Planning, Perception and Navigation for Intelligent Vehicles*, Tokyo, Japan, 2013.
- [62] M. Mazzola and G. Schaaf, “Modeling and control design of a centralized adaptive cruise control system,” *International Journal of Computer, Electrical, Automation, Control and Information Engineering*, vol. 8, pp. 1142–1146, 2014.
- [63] R.-H. Gielen, S. Olaru, M. Lazar, W.-P.-M.-H. Heemels, N. van de Wouw, and S.-I. Niculescu, “On polytopic inclusions as a modeling framework for systems with time-varying delays,” *Automatica*, vol. 46, pp. 615–619, 2010.
- [64] K. Natori, R. Oboe, and K. Ohnishi, “Stability analysis and practical design procedure of time delayed control systems with communication disturbance observer,” *IEEE Transactions on Industrial Informatics*, vol. 4, pp. 185–197, 2008.

- [65] K. Natori, T. Tsuji, K. Ohnishi, A. Hace, and K. Jezernik, "Time-delay compensation by communication disturbance observer for bilateral teleoperation under time-varying delay," *IEEE Transactions on Industrial Electronics*, vol. 57, pp. 1050–1061, 2010.
- [66] C. F. Caruntu and C. Lazar, "Robustly stabilising model predictive control design for networked control systems with an application to direct current motors," *IET Control Theory and Applications*, vol. 6, pp. 943–952, 2012.
- [67] ———, "Stabilizing MPC for network-controlled systems with an application to DC motors," in *IEEE International Conference on Mechatronics*, Istanbul, Turkey, 2011, pp. 973–978.
- [68] M. Lazar and W. P. M. H. Heemels, "Optimized input-to-state stabilization of discrete-time nonlinear systems with bounded inputs," in *27th American Control Conference*, Seattle, USA, 2008.
- [69] T. A. Johansen, "Toward dependable embedded model predictive control," *IEEE Systems Journal*, 2014.
- [70] B. Huyck, L. Callebaut, F. Logist, H. J. Ferreau, M. Diehl, J. De Brabanter, J. Van Impe, and B. De Moor, "Implementation and experimental validation of classic MPC on programmable logic controllers," in *20th Mediterranean Conference on Control & Automation*, Spain, 2013, pp. 679–684.
- [71] L. G. Bleris, P. D. Vouzis, M. G. Arnold, and M. V. Kothare, "A co-processor FPGA platform for the implementation of real-time model predictive control," in *American Control Conference*, Minnesota, USA, 2006.
- [72] G. Knagge, A. Wills, A. Mills, and B. Ninness, "ASIC and FPGA implementation strategies for model predictive control," in *European Control Conference*, Budapest, Hungary, 2009, pp. 144–149.
- [73] A. Wills, A. Mills, and B. Ninness, "FPGA implementation of an interior-point solution for Linear Model Predictive Control," in *18th IFAC World Congress*, Milano, Italy, 2011.
- [74] A. Wills, G. Knagge, and B. Ninness, "Fast linear model predictive control via custom integrated circuit architecture," *IEEE Transactions on Control Systems Technology*, vol. 20, pp. 50–71, 2012.
- [75] N. Yang, D. Li, J. Zhang, and Y. Xi, "Model predictive controller design and implementation on FPGA with application to motor servo system," *Control Engineering Practice*, vol. 20, pp. 1229–1235, 2012.
- [76] J. L. Jerez, P. J. Goulart, S. Richter, G. A. Constantinides, E. C. Kerrigan, and M. Morari, "Embedded online optimization for model predictive control at megahertz rates," *IEEE Transactions on Automatic Control*, vol. 59, pp. 3238–3251, 2014.

- [77] E. N. Hartley and J. M. Maciejowski, “Graphical FPGA Design for a Predictive Controller with Application to Spacecraft Rendezvous,” in *52nd Annual Conference on Decision and Control*, Italy, 2013, pp. 1971–1976.
- [78] ———, “Predictive control for spacecraft rendezvous in an elliptical orbit using an FPGA,” in *European Control Conference*, Zurich, Switzerland, 2013, pp. 1359–1364.
- [79] E. N. Hartley, J. L. Jerez, A. Suardi, J. M. Maciejowski, E. C. Kerrigan, and G. A. Constantinides, “Predictive control using an FPGA with application to aircraft control,” *IEEE Transactions on Control Systems Technology*, vol. 22, pp. 1006–1017, 2014.
- [80] J. Liu, H. Peyrly, A. Burgz, and G. A. Constantinides, “FPGA implementation of an interior point method for high-speed model predictive control,” in *24th International Conference on Field Programmable Logic and Applications*, Munich, Germany, 2014.
- [81] K. Mohamed, A. El Mahdy, and M. Refai, “Model predictive control using FPGA,” *International Journal of Control Theory and Computer Modeling*, vol. 5, 2015.
- [82] A. A. Kheriji, F. Bouani, and M. Ksouri, “A Microcontroller Implementation of Constrained Model Predictive Control,” *International Journal of Electrical and Electronics Engineering*, 2011.
- [83] A. A. Kheriji, F. Bouani, M. Ksouri, and M. B. Ahmed, “A Microcontroller Implementation of Constrained Model Predictive Control,” *International Journal of Electrical, Computer, Energetic, Electronic and Communication Engineering*, 2011.
- [84] P. Matousek, “Microprocessor system designated for control pneumatic actuator,” *Journal of Applied Science in the Thermodynamics and Fluid Mechanics*, vol. 5, 2011.
- [85] J. Currie, A. Prince-Pike, and D. I. Wilson, “Auto-code generation for fast embedded model predictive controllers,” in *19th International Conference on Mechatronics and Machine Vision in Practice*, New-Zealand, 2012.
- [86] K.-J. Barsk, “Model Predictive Control of a Tricopter,” Ph.D. dissertation, PhD Thesis, Department of Electrical Engineering, Linköping University, Sweden, 2012.
- [87] B. P. Nguyen, Y. Ho, Z. Wu, and C.-K. Chui, “Implementation of model predictive control with modified minimal model on low-power RISC microcontrollers,” in *3rd Symposium on Information and Communication Technology*, Vietnam, 2012, pp. 165–171.
- [88] K.-J. Barsk, “On-board Model Predictive Control of a Quadrotor Helicopter: Design, Implementation, and Experiments, technical report,” Ph.D. dissertation, Electrical Engineering and Computer Sciences University of California at Berkeley, 2012.

- [89] M. Bangura and R. Mahony, “Real-time Model Predictive Control for Quadrotors,” in *19th IFAC World Congress*, Cape Town, South Africa, 2014.
- [90] T. Baca, “Model Predictive Control of Micro Aerial Vehicle Using Onboard Micro-controller,” Ph.D. dissertation, Masters Thesis, Faculty of Electrical Engineering, Czech Technical University in Prague, 2015.
- [91] J. Novak and P. Chalupa, “Implementation of Fuzzy MPC on Embedded System,” *Latest Trends on Systems*, 2014.
- [92] J. L. Jerez, G. A. Constantinides, and E. C. Kerrigan, “Towards a fixed point QP solver for predictive control,” in *51st IEEE Conference on Decision and Control*, USA, 2012.
- [93] G. E. and J. F.A., “Experiments with vehicle platooning,” Ph.D. dissertation, Masters Thesis in Embedded and Intelligent Systems, School of Information Science, Computer and Electrical Engineering, Halmstad University, 2010.
- [94] D. Martinec, “Distributed control of platoons of racing slot cars,” Ph.D. dissertation, Diploma Thesis, Department of Control Engineering, Faculty of Electrical Engineering, Czech Technical University in Prague, 2012.
- [95] A. Keskikangas and G. Sallberg, “Designing and implementing a model vehicle platoon with longitudinal control,” Ph.D. dissertation, Masters Thesis, Department of Automatic Control, Lund University, Sweden, 2014.
- [96] T. A. Johansen, W. Jackson, R. Schrieber, and P. Tondel, “Hardware Architecture Design for Explicit Model Predictive Control,” in *American Control Conference*, Minneapolis, USA, 2006.
- [97] R. Rajamani, H.-S. Tan, B. K. Law, and W.-B. Zhang, “Demonstration of integrated longitudinal and lateral control for the operation of automated vehicles in platoons,” *IEEE Transactions on Control Systems Technology*, pp. 695–708, 2000.
- [98] A. Bemporad, F. Borrelli, and M. Morari, “The explicit solution of constrained LP-based receding horizon control,” in *39th IEEE Conf.on Decision and Control*, 2000, pp. 632–637.
- [99] E. Gilbert and K. Tan, “Linear systems with state and control constraints: The theory and application of maximal output admissible sets,” *IEEE Transactions on Automatic Control*, vol. 9, pp. 1008–1020, 1991.
- [100] J. B. Rawlings and D. Q. Mayne, *Model Predictive Control: Theory and Design*. Nob Hill Publishing, 2009.
- [101] J. M. Maciejowski, *Predictive Control with Constraints*. Prentice Hall, 2000.
- [102] W. Dunbar, “Distributed receding horizon control of dynamically coupled nonlinear systems,” *IEEE Transactions on Automatic Control*, vol. 52, no. 7, pp. 1249–1263, 2007.

- [103] Y. Kuwata, A. Richards, and J. How, “Robust receding horizon control for large scale dynamically decoupled systems,” *Automatica*, vol. 42, no. 12, pp. 2105–2115, 2006.
- [104] J. Dolt and O. Stursberg, “Distributed predictive control of communicating and platooning vehicles,” in *Joint 48th IEEE Conference on Decision and Control and 28th Chinese Control Conference*, Shanghai, P.R. China, 2009, pp. 561–566.
- [105] Y. Hong, X. Wang, and Z.-P. Jiang, “Distributed output regulation of leader-follower multi-agent systems,” *Int. J. Robust Nonlinear Control*, vol. 23, no. 1, pp. 48–66, 2013.
- [106] H. Fukushima, K. Kon, and F. Martumo, “Distributed model predictive control for multi-vehicle formation with collision avoidance constraints,” in *44th IEEE Conference on Decision Control and the European Control Conference*, Seville, Spain, 2000, pp. 5480–5485.
- [107] C. Jian, D. Sun, J. Yang, and H. Chen, “Leader-follower formation control of multiple non-holonomic mobile robots incorporating a receding-horizon scheme,” *International Journal of Robotics Research*, pp. 727–747, 2010.
- [108] Z. Li, W. Ren, X. Liu, and M. Fu, “Distributed containment control of multi-agent systems with general dynamics in the presence on multiple leaders,” *Int. J. Robust. Nonlinear Control*, pp. 534–547, 2013.
- [109] V. Gupta, B. Hassibi, and R. Murray, “A sub-optimal algorithm to synthesize control laws for a network of dynamic agents,” *International Journal of Control*, vol. 16, pp. 1302–1313, 2005.
- [110] H. Li and Y. Shi, “Constrained continuous-time nonlinear systems: A robustness constrained Approach,” *IEEE Tran. Autom. Control*, pp. 1673–1678, 2014.
- [111] F. A. C. C. Fontes, D. B. M. M. Fontes, and A. C. D. Caldeira, “Model predictive control of vehicle formations,” *Lecture notes in control and information sciences*, Springer Berlin Heidelberg, pp. 371–384, 2009.
- [112] B. J. Young, R. W. Beard, and J. M. Kelsey, “A control scheme for improving multi-vehicle formation maneuvers,” in *American Control Conference*, Arlington, VA, USA, 2001, pp. 704–709.
- [113] A. Venkat, J. Rawlings, and S. Wright, “Distributed model predictive control of large-scale systems,” *LNCIS*, vol. 358, no. 5, pp. 591–605, 2007.
- [114] P. Christofides, R. Scattolini, D. Munoz de la Pena, and J. Liu, “Distributed model predictive control: A tutorial review and future research directions,” *Computers and Chemical Engineering*, vol. 51, no. -, pp. 21–41, 2013.
- [115] R. Scattolini, “Architectures for distributed and hierarchical model predictive control a review,” *Journal of Process Control*, vol. 19, no. 5, pp. 723–731, 2009.

- [116] U. Maeder and M. Morari, “Offset-free reference tracking for predictive controllers,” in *46th IEEE Conference on Decision and Control*, New Orleans, USA, 2007, pp. 5252–5257.
- [117] A. Faanes and S. Skogestad, “Offset-free tracking with mpc with model mismatch: Experimental results,” *Industrial & Engineering Chemical Research*, vol. 44, no. 11, pp. 3966–3972, 2005.
- [118] F. Borrelli and M. Morari, “Offset free model predictive control,” in *46th IEEE Conference on Decision and Control*, New Orleans, USA, 2007, pp. 1245–1250.
- [119] G. Betti, “Methods and applications of distributed and decentralized model predictive control,” Ph.D. dissertation, Politecnico di Milano, 2013.
- [120] G. Pannocchia and J. Rawlings, “Disturbance models for offset-free model predictive control,” *AIChE Journal*, vol. 49, pp. 426–437, 2003.
- [121] L. Magni and R. Scattolini, “Tracking of non-square nonlinear continuous time systems with piecewise constant model predictive control,” *Journal of Process Control*, vol. 17, no. 8, pp. 631–640, 2007.
- [122] L. Wang, *Control System Design and Implementation Using MATLAB*. Springer-Verlag, 2009.
- [123] A. Maxim, C. Lazar, and C. Caruntu, “A computationally efficient non-cooperative distributed model predictive control algorithm for two agent systems,” in *20th International Conference on Control Systems and Computer Science*, Bucharest, Romania, 2015, pp. 673 – 678.
- [124] W. Ren and Y. Cao, “Overview of recent research in distributed multi-agent coordination,” *Distributed Coordination of Multi-agent Networks*, Springer-Verlag, pp. 23 – 41, 2011.
- [125] M. Ji and M. Egerstedt, “Distributed coordination control of multiagent systems while preserving connectedness,” *IEEE Trans. on Robot.*, vol. 23, no. 4, pp. 693–703, 2007.
- [126] D. Panagou and V. Kumar, “Cooperative visibility maintenance for leader-follower formations in obstacle environments,” *IEEE Trans. on Robot.*, vol. 30, no. 4, pp. 831–844, 2014.
- [127] D. Panagou, D. Stipanovic, and P. Vulgaris, “Distributed coordination control for multi-robot networks using lyapunov-like barrier functions,” *IEEE Trans. on Automatic Control*, vol. 61, no. 3, pp. 617–632, 2016.
- [128] W. Dunbar and R. Murray, “Model predictive control of coordinated multi-vehicle formations,” in *41st IEEE Conference on Decision and Control*, Las Vegas, USA, 2002, pp. 4631–4636.

- [129] R. Olfati-Saber, “Distance-based undirected formations of single-integrator and double-integrator modeled agents in n-dimensional space,” *IEEE TRANS AUTOMAT CONTR*, vol. 51, no. 3, pp. 401–420, 2006.
- [130] H. Tanner, A. Jadbabaie, and G. Pappas, “Flocking in fixed and switching networks.” *IEEE IEEE TRANS AUTOMAT CONTR*, vol. 52, no. 5, pp. 287–292, 2007.
- [131] X. Cai and M. de Queiroz, “Rigidity-based stabilization of multi-agent formations.” *Journal of Dynamic Systems Measurement and Control*, vol. 136, no. 1, pp. 0 145 021 – 0 145 027, 2013.
- [132] —, “Multi-agent formation maneuvering and target interception with double-integrator model.” in *2014 American Control Conference*, Portland, USA, 2014, pp. 287 – 292.
- [133] K.-K. Oh and H.-S. Ahn, “Distance-based undirected formations of single-integrator and double-integrator modeled agents in n-dimensional space,” *Int. J. Robust Non-linear Control*, vol. 24, no. 12, pp. 1809–1820, 2014.
- [134] G. Franklin, A. Emami-Naeini, and J. Powell, *Feedback Control of Dynamic Systems*, 3rd. Addison-Wesley Longman Publishing Co., 1993.
- [135] C. Caruntu, M. Lazar, R. Gielen, and P. J. van den Bosch, “Lyapunov based predictive control of vehicle drivetrains over can,” *Control Engineering Practice*, vol. 21, no. -, pp. 1884–1898, 2013.
- [136] G. Orosy and G. F. Bullo, “Delayed car-following dynamics for human and robotic driver,” in *ASME 2011 International Design Engineering Technical Conferences & Computers and Information in Engineering Conference*, Washington, USA, 2011, pp. 529–538.
- [137] A. Maxim, C. F. Caruntu, and C. Lazar, “Cruise and headway control for vehicle platooning using a distributed model predictive control algorithm,” Sinaia, Romania, 2017.
- [138] A. G. Ulsoy, H. Peng, and M. Cakmakci, *Automotive Control Systems*. Editura Cambridge University Press, 2012.
- [139] J. Zhou and H. Peng, “Range policy of adaptive cruise control vehicles for improved flow stability and string stability,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 6, no. 2, pp. 229 – 237, 2005.
- [140] J. Zhang, F.-Y. Wang, K. Wang, W.-H. Lin, X. Xu, and C. Chen, “Data-driven intelligent transportation systems: A survey,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 12, no. 4, pp. 1624 – 1639, 2011.
- [141] C. F. Caruntu, C. Braescu, A. Maxim, R. C. Rafaila, and A. Tiganasu, “Distributed model predictive control for vehicle platooning: A brief survey,” Sinaia, Romania, 2016, pp. 644 – 650.

- [142] Y. Zheng, S. E. Li, K. Li, F. Borrelli, and J. K. Hedrick, “Distributed model predictive control for heterogeneous vehicle platoons under unidirectional topologies,” *IEEE Transactions on Control Systems Technology*, vol. 25, no. 3, pp. 899 – 910, 2017.
- [143] X. Qian, A. La Fortelle, and F. Moutarde, “A Hierarchical Model Predictive Control Framework for On-road Formation Control of Autonomous Vehicles,” in *IEEE Intelligent Vehicle Symposium*, Goteborg, Sweden, June 2016.
- [144] L. Dai, Q. Cao, Y. Xian, and Y. Gao, “Distributed mpc for formation of multi-agent systems with collision avoidance and obstacle avoidance,” *Journal of the Franklin Institute*, vol. 354, pp. 2068 – 2085, 2017.
- [145] V. Turri, B. Besselink, and K. H. Johansson, “Cooperative look-ahead control for fuel-efficient and safe heavy-duty vehicle platooning,” *IEEE Transactions on Control Systems Technology*, vol. 25, no. 1, pp. 12 – 28, 2017.
- [146] A. Maxim, C. F. Caruntu, and C. Lazar, “Distributed model predictive control algorithm for vehicle platooning,” Sinaia, Romania, 2016, pp. 657 – 662.
- [147] M. K. Liubakka, D. S. Rhode, J. R. Winkelman, and P. V. Kokotovic, “Adaptive automotive speed control,” *IEEE Transactions on Automatic Control*, vol. 38, no. 7, pp. 1011 – 1020, 1993.
- [148] K. Oda, H. Takeuchi, M. Tsujii, and M. Ohba, “Practical estimator for self-tuning automotive cruise control,” Boston, MA, USA, 1991, pp. 2066 – 2071.
- [149] M. Tsujii, H. Takeuchi, K. Oda, and M. Ohba, “Application of Self-Tuning to Automotive Cruise Control,” in *American Control Conference*, San Diego, CA, 23rd-25th May 1990, pp. 1843–1848.
- [150] J. Lofberg, “Yalmip : A toolbox for modeling and optimization in matlab,” New Orleans, LA, USA, 2004, pp. 284 – 289.
- [151] A. Maxim, J. M. Maestre, C. F. Caruntu, and C. Lazar, “A methodology to establish the disturbance boundaries for a min-max coalitional model predictive control framework,” Limassol, Cyprus, 2018.
- [152] ———, “Min-max coalitional model predictive control algorithm for multiple agents,” Copenhagen, Denmark, 2018.
- [153] J. H. Lee and Z. Yu, “Worst-case formulations of model predictive control for systems with bounded parameters,” *Automatica*, vol. 33, no. 5, pp. 763–781, 1997.
- [154] A. Bemporad and M. Morari, *Robustness in identification and control. Lecture notes in Control and Information Sciences, vol 245*. Springer, London, 1999.
- [155] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, “Constrained model predictive control: Stability and optimality,” *Automatica*, vol. 36, no. 6, pp. 789–814, 2000.

- [156] J. C. Allwright and G. C. Papavasiliou, “On linear programming and robust model predictive control using impulse-responses,” *Systems & Control Letters*, vol. 18, no. 2, pp. 159–164, 1992.
- [157] E. C. Kerrigan and J. M. Maciejowski, “Feedback min-max model predictive control using a single linear program: Robust stability and the explicit solution,” *International Journal of Robust and Nonlinear Control*, vol. 14, no. 4, pp. 395–413, 2004.
- [158] H. S. Witsenhausen, “A minimax control problem for sampled linear systems,” *IEEE Transactions on Automatic Control*, vol. 13, no. 1, pp. 5–21, 1968.
- [159] D. A. Coop and J. P. Hespanha, “Nonlinear output-feedback model predictive control with moving horizon estimation,” Los Angeles, CA, USA, 2014, pp. 3511–3517.
- [160] D. Munoz de la Pena, T. Alamo, D. Ramirez, and E. F. Camacho, “Min-max model predictive control as a quadratic program,” *IFAC Proceedings Volumes*, vol. 38, no. 1, pp. 263–268, 2005.
- [161] M. Diehl, “Formulation of a closed-loop min-max mpc as a quadratic constrained program,” *IEEE Transactions on Automatic Control*, vol. 52, no. 2, pp. 339–343, 2007.
- [162] J. Lofberg, “Aproximations of closed-loop minimax mpc,” Maui, Hawaii, USA, 2003, pp. 1438–1442.
- [163] A. Bemporad, F. Borrelli, and M. Morari, “Min-max control of constrained uncertain discrete-time linear systems,” *IEEE Transactions on Automatic Control*, vol. 48, no. 9, pp. 1600–1606, 2003.
- [164] D. Limon, T. Alamo, F. Salas, and E. F. Camacho, “Input to state stability of min-max mpc controllers for nonlinear systems with bounded uncertainties,” *Automatica*, vol. 42, no. 5, pp. 797–803, 2006.
- [165] D. M. Raimondo, L. Magni, and R. Scattolini, “Decentralized open-loop mpc of nonlinear systems: an input-to-state stability approach,” Kos, Greece, 2007, pp. 1642–1647.
- [166] J. M. Maestre and R. R. Negenborn, *Distributed Model Predictive Control Made Easy*. Springer, 2013.
- [167] D. Jia and B. Krogh, “Min-max feedback model predictive control for distributed control with communication,” Anchorage, AK, USA, 2002, pp. 4507–4512.
- [168] F. Fele, J. M. Maestre, and E. F. Camacho, “Coalitional control: Cooperative game theory and control,” *IEEE Control Systems Magazine*, vol. 37, no. 1, pp. 53–69, 2017.
- [169] J. M. Maestre, D. Munoz de la Pena, A. J. Losada, E. Algaba, and E. Camacho, “A coalitional control scheme with applications to cooperative game theory,” *Optimal Control Applications and Methods*, vol. 35, no. 5, pp. 592–608, 2014.

- [170] M. Jilg and O. Stursberg, “Hierarchical distributed control for interconnected systems,” Shanghai, China, 2013, pp. 419–425.
- [171] F. Fele, J. M. Maestre, M. Hashemy, D. Munoz de la Pena, and E. F. Camacho, “Coalitional model predictive control of an irrigation canal,” *Journal of Process Control*, vol. 24, no. 4, pp. 314–325, 2014.
- [172] J. M. Maestre and H. Ishii, “A page rank based coalitional control scheme,” *International Journal of Control, Automation and Systems- In press*, 2018.
- [173] P. Trodden and A. Richards, “Adaptive cooperation in robust distributed model predictive control,” Saint Petersburg, Russia, 2009, pp. 896–901.
- [174] S. Lucia, M. Kogel, and R. Findeisen, “Adaptive cooperation in robust distributed model predictive control,” *IFAC-PapersOnLine*, vol. 48, no. 23, pp. 205–211, 2015.
- [175] J. M. Maestre, D. Munoz de la Pena, and E. F. Camacho, “Distributed model predictive control based on a cooperative game,” *Optimal Control Applications and Methods*, vol. 32, no. 2, pp. 153–176, 2011.
- [176] J. Lofberg, “Automatic robust convex programming,” *Optimization Methods and Software*, vol. 27, no. 1, pp. 115–129, 2010.
- [177] S. Chakraborty, M. A. Al Faruque, W. Chang, D. Goswami, M. Wolf, and Q. Zhu, “Automotive Cyber-Physical Systems: A Tutorial Introduction,” *IEEE Design & Test*, pp. 92–108, 2016.
- [178] D. Jia, K. Lu, J. Wang, X. Zhang, and X. Shen, “A Survey on Platoon-Based Vehicular Cyber-Physical Systems,” *IEEE Communication Surveys & Tutorials*, vol. 18, no. 1, pp. 263–284, 2016.
- [179] D. Goswami, R. Schneider, A. Masrur, M. Lukasiewicz, S. Chakraborty, H. Voit, and A. Annaswamy, “Challenges in Automotive Cyber-physical Systems Design,” in *International Conference on Embedded Computer Systems*, 16th-19th July 2012.
- [180] Y. Zheng, S. E. Li, K. Li, F. Borrelli, and J. K. Hedrick, “Distributed Model Predictive Control for Heterogeneous Vehicle Platoons under Unidirectional Topologies,” *IEEE Transaction on Control Systems Technology*, 2016.
- [181] L. D. Baskar, B. De Shutter, and H. Hellendoorn, “Model Predictive Control for Intelligent Speed Adaptation in Intelligent Vehicle Highway Systems,” in *17th IEEE International Conference on Control Applications*, San Antonio, Texas, 2008, pp. 468–473.
- [182] Y. Cao and F. Yu, “Generalized Predictive Control based on Vehicle Path Following Strategy by using Active Steering System,” in *13th International Symposium on Advanced Vehicle Control*, Munchen, Germany, 13th-16th September 2016.
- [183] A. Pawlowski, J. L. Guzman, J. E. Normey-Rico, and M. Berenguel, “Improving feedforward disturbance compensation capabilities in Generalized Predictive Control,” *Journal Process Control*, vol. 22, pp. 527–539, 2012.

- [184] E. F. Camacho and C. Bordons, *Model Predictive Control*. Springer-Verlag, London, UK, 2007.
- [185] K. C. Dey, L. Yan, X. Wang, Y. Wang, H. Shen, M. Chowdhury, L. Yu, and C. Qiu, “A Review of Communication, Driver Characteristics and Controls Aspects of Cooperative Adaptive Cruise Control (CACC),” *IEEE Transactions On Intelligent Transportation Systems*, vol. 17, pp. 491–509, 2015.
- [186] S. E. Shladover, C. Nowakowski, X.-Y. Lu, and R. Ferlis, “Cooperative Adaptive Cruise Control (CACC) Definitions And Operating Concepts,” in *94th TRB Annual Conference*, Washington D. C., USA, 2015.
- [187] D. A. Farias, “Using Adaptive and Cooperative Adaptive Cruise Control to Maximize Throughput of Signalized Arterials, Technical Report,” University of California at Berkeley, 2017.
- [188] “Transportation research board. highway capacity manual.” in *Washington, DC, USA*, 2017.
- [189] J. Lioris, R. Pedarsani, F. Y. Tascikaraoglu, and P. Varaiya, “Platoons of connected vehicles can double throughput in urban roads,” *Transportation Research Part C: Emerging Technologies*, vol. 77, pp. 292–305, 2012.
- [190] J. Ploeg, B. T. M. Scheepers, and E. van Nunen, “Design and experimental evaluation of cooperative adaptive cruise control,” in *IEEE International Conference on Intelligent Transportation Systems*, Washington D.C., 2011, pp. 5–7.
- [191] M. Treiber, A. Hennecke, and D. Helbing, “Congested traffic states in empirical observations and microscopic simulations,” *Physical Review E*, vol. 62(2), pp. 1805–1824, August 2000.
- [192] P. Falcone, M. Tufo, F. Borrelli, J. Asgari, and H. E. Tseng, “Linear time varying model predictive control approach to the integrated vehicle dynamics control problem in autonomous systems,” in *IEEE Conference on Decision and Control*, New Orleans, Los Angeles, USA, 2007, pp. 2980–2985.
- [193] T. Besselmann and M. Morari, “Hybrid parameter-varying model predictive control for autonomous vehicle steering,” *European Journal of Control*, vol. 5, pp. 418–431, 2008.
- [194] A. Carvalho, S. Lefevre, G. Schildbach, J. Kong, and F. Borrelli, “Automated driving: The role of forecasts and uncertainty - a control perspective,” *European Journal of Control*, vol. 24, pp. 14–32, 2015.
- [195] R. C. Rafaila and G. Livint, “Predictive control of autonomous steering for ground vehicles,” in *9th International Symposium on Advanced Topics in Electrical Engineering*, Bucharest, Romania, 2015, pp. 543–547.
- [196] —, “Nonlinear model predictive control of autonomous vehicle steering,” in *19th International Conference on System Theory, Control and Computing*, Cheile Gradistei, Romania, 2015, pp. 466–471.

- [197] R. C. Rafaila, C. F. Caruntu, and G. Livint, “Nonlinear model predictive control using Lyapunov functions for vehicle lateral dynamics,” in *14th IFAC Symposium on Control in Transportation Systems*, Istanbul, Turkey, 2016.
- [198] R. C. Rafaila, G. Livint, and F. A. Rusu, “Multivariable nonlinear predictive control of autonomous vehicle dynamics,” in *13th International Conference on Development And Application Systems*, Suceava, Romania, 2016.
- [199] A. Katriniok, J. P. Maschuw, F. Christen, L. Eckstein, and D. Abel, “Optimal vehicle dynamics control for combined longitudinal and lateral autonomous vehicle guidance,” in *European Control Conference*, Zurich, Germany, 2013.
- [200] R. Attia, R. Orjuela, and M. Basset, “Combined longitudinal and lateral control for automated vehicle guidance,” *Vehicle System Dynamics*, vol. 52, pp. 261–279, 2014.
- [201] A. Bacha, C. Bauman, R. Faruque, M. Fleming, C. Terwelp, C. Reinholtz, D. Hong, A. Wicks, T. Alberi, D. Anderson, and et al., “Odin: Team VictorTangos entry in the DARPA Urban Challenge,” *Journal of Field Robotics*, 2008.
- [202] B. Donald, P. Xavier, J. Canny, and J. Reif, “Kinodynamic motion planning,” *Journal of the ACM*, 1994.
- [203] L. E. Dubins, “On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents,” *American Journal of Mathematics*, 1957.
- [204] A. H. Glattfelder, W. Schaufelberger, and H. Fassler, “Stability of override control systems,” *International Journal of Control*, 1983.
- [205] P. Fiorini and Z. Shiller, “Time optimal trajectory planning in dynamic environments,” in *IEEE International Conference on Robotics and Automation*, 1996.
- [206] L. Fletcher, S. Teller, E. Olson, D. Moore, Y. Kuwata, J. How, J. Leonard, I. Miller, M. Campbell, and D. Huttenlocher, “The MIT Cornell collision and why it happened,” *Journal of Robotic Systems*, 2008.
- [207] A. Kelly and B. Nagy, “Reactive nonholonomic trajectory generation via parametric optimal control,” *International Journal of Robotics Research*, 2003.
- [208] Y. Kuwata, G. A. Fiore, J. Teo, E. Frazzoli, and J. P. How, “Motion planning for urban driving using RRT,” in *International Conference on Intelligent Robots and Systems*, 2008.
- [209] J. Z. Kolter, C. Plagemann, D. T. Jackson, A. Y. Ng, and S. Thrun, “A probabilistic approach to mixed open-loop/closed-loop control, with application to extreme autonomous driving,” in *International Conference on Robotics and Automation*, 2010.
- [210] V. Delsart, T. Fraichard, and L. Martinez, “Real-time trajectory generation for car-like vehicles navigating dynamic environments,” in *IEEE International Conference on Robotics and Automation*, 2009, pp. 3401–3406.

- [211] O. Sawodny, J. Zimmermann, and A. Lutz, "Motion planning for an autonomous vehicle driving on motorways by using flatness properties," in *IEEE International Conference on Control Applications*, 2010, pp. 908–913.
- [212] L. Ma, J. Yang, and M. Zhang, "A two-level path planning method for on-road autonomous driving," in *2012 2nd International Conference on Intelligent System Design and Engineering Application*, 2012, pp. 661–664.
- [213] J. Perez, J. Godoy, J. Villagra, and E. Onieva, "Trajectory generator for autonomous vehicles in urban environments," in *IEEE International Conference on Robotics and Automation*, 2013, pp. 409–414.
- [214] D. Gonzalez, J. Perez, R. Lattarulo, V. Milanese, and F. Nashashibi, "Continuous curvature planning with obstacle avoidance capabilities in urban scenarios," in *17th International IEEE Conference on Intelligent Transportation Systems*, 2014, pp. 1430–1435.
- [215] I. Bae, J. Moon, H. Park, J. H. Kim, and S. Kim, "Path generation and tracking based on a Bezier curve for a steering rate controller of autonomous vehicles," in *IEEE Conference on Intelligent Transportation Systems*, 2013, pp. 436–441.
- [216] M. Wang, T. Ganjineh, and R. Rojas, "Action annotated trajectory generation for autonomous maneuvers on structured road networks," in *5th International Conference on Automation, Robotics and Applications*, 2011, pp. 67–72.
- [217] T. Gu and J. M. Dolan, "Toward human-like motion planning in urban environments," in *IEEE Intelligent Vehicles Symposium*, 2014, pp. 350–355.
- [218] J. Ziegler, P. Bender, T. Dang, C. Stiller, and A. Prelimaries, "Trajectory planning for BERTHA a local, continuous method," in *IEEE Intelligent Vehicles Symposium (IV)*, 2014, pp. 450–457.
- [219] D. Madas, M. Nosratinia, M. Keshavarz, P. Sundstrom, R. Philippsen, A. Eidehall, and K. Dahl, "On path planning methods for automotive collision avoidance," in *2013 IEEE Intelligent Vehicles Symposium (IV)*, 2013, pp. 931–937.
- [220] X. Li, Z. Sun, Q. Zhu, and D. Liu, "A unified approach to local trajectory planning and control for autonomous driving along a reference path," in *International Conference on Mechatronics and Automations*, 2014, pp. 1716–1721.
- [221] M. Werling, J. Ziegler, S. Kammel, and S. Thrun, "Optimal trajectory generation for dynamic street scenarios in a Frenet Frame," in *IEEE International Conference on Robotics and Automation*, Anchorage, Alaska, USA, 2010.
- [222] C. Rathgeber, F. Winkler, X. Kang, and S. Muller, "Optimal trajectories for highly automated driving," *International Journal of Mechanical, Aerospace, Industrial, Mechatronic and Manufacturing Engineering*, 2015.
- [223] M. LaValle, "Motion planning for urban driving using RRT," Ph.D. dissertation, Technical report, Computer Science Dept., Iowa State University, 1998.

- [224] A. T. Schwarm and M. Nikolaou, “Chance-constrained model predictive control,” *AIChE Journal*, vol. 45, pp. 1743–1752, 1999.
- [225] P. Li, M. Wendt, and G. Wozny, “A probabilistically constrained model predictive controller,” *Automatica*, vol. 38, pp. 1171–1176, 2002.
- [226] J. Yan and R. R. Bitmead, “Incorporating state estimation into model predictive control and its application to network traffic control,” *Automatica*, vol. 41, pp. 595–604, 2005.
- [227] I. S. Choi, A. Rossiter, and P. Fleming, “Effectiveness of MPC algorithms for hot rolling mills in the presence of disturbances,” in *American Control Conference*, New York City, USA, 2007, pp. 4118–4123.
- [228] L. Blackmore and M. Ono, “Convex chance constrained predictive control without sampling,” in *AIAA Guidance, Navigation and Control Conference*, Chicago, IL, USA, 2009.
- [229] M. Cannon, B. Kouvaritakis, S. Rakovic, and Q. Cheng, “Stochastic tubes in model predictive control with probabilistic constraints,” *IEEE Transactions on Automatic Control*, vol. 56, pp. 194–200, 2011.
- [230] D. Bernardini and A. Bemporad, “Stabilizing model predictive control of stochastic constrained linear systems,” *IEEE Transactions on Automatic Control*, vol. 57, pp. 1468–1480, 2012.
- [231] J. M. Grosso, J. M. Maestre, C. Ocampo-Martinez, and V. Puig, “On the assessment of tree-based and chance-constrained predictive control approaches applied to drinking water networks,” in *19th IFAC World Congress*, Cape Town, South Africa, 2014, pp. 6240–6245.
- [232] M. Farina, L. Giulioni, L. Magni, and R. Scattolini, “Output feedback model predictive control: a probabilistic approach,” in *19th IFAC World Congress*, Cape Town, South Africa, 2014, pp. 7461–7466.
- [233] S. Yu, T. Qu, F. Xu, and H. Chen, “Model predictive control of linear stochastic systems with constraints,” in *American Control Conference*, Chicago, IL, USA, 2015, pp. 950–955.
- [234] M. Farina, L. Giulioni, L. Magni, and R. Scattolini, “An approach to output-feedback MPC of stochastic linear discrete-time systems,” *Automatica*, vol. 55, pp. 140–149, 2015.
- [235] M. Farina and R. Scattolini, “Model predictive control of linear systems with multiplicative unbounded uncertainty and chance constraints,” *Automatica*, vol. 70, pp. 258–265, 2016.
- [236] C. F. Caruntu, C. C. Velandia-Cardenas, X. Liu, and A. N. Vargas, “Model predictive control with probability constraints: application to networked systems,” *submitted to International Journal of Robust and Nonlinear Systems*, 2017.

- [237] E. I. Verriest and W. Michiels, “Stability analysis of systems with stochastically varying delays,” *Systems & Control Letters*, vol. 58, pp. 783–791, 2009.
- [238] M. B. G. Cloosterman, N. van den Wouw, W. P. M. H. Heemels, and H. Nikmeijer, “Stabilization of networked control systems with large delays and packet dropouts,” in *American Control Conference*, Seattle, WA, USA, 2008, pp. 4991–4996.
- [239] C. Tan, L. Li, and H. Zhang, “Stabilization of networked control systems with both network-induced delay and packet dropout,” *Automatica*, vol. 59, pp. 194–199, 2015.
- [240] M. Yu, L. Wang, T. Chu, and F. Hao, “Stabilization of networked control systems with data packet dropout and transmission delays: Continuous-time case,” *European Journal of Control*, vol. 11, pp. 40–49, 2005.
- [241] X. Guan, S. Dai, and C. Long, “Controller design for networked control system with data packet dropout and transmission delays,” *Journal of Control Theory and Applications*, vol. 5, pp. 227–232, 2007.
- [242] J. Gao, M. Fu, and J. Ren, “Output feedback stabilization of wireless networked control system with packet dropout,” in *10th IEEE International Conference on Control and Automation*, Hangzhou, China, 2013, pp. 1696–1701.
- [243] H. Shao, Q.-L. Han, J. Zhao, and D. Zhang, “A separation method of transmission delays and data packet dropouts from a lumped input delay in the stability problem of networked control systems,” *International Journal of Robust and Nonlinear Control*, 2016.
- [244] C. F. Caruntu, “Observer-based predictive controller design with network-enhanced time-delay compensation,” *International Journal of General Systems*, vol. 44, pp. 182–197, 2015.
- [245] M. Lazar, “Flexible control Lyapunov functions,” in *28th American Control Conference*, St. Louis, MO, USA, 2009, pp. 102–107.
- [246] Z.-P. Jiang and Y. Wang, “Input-to-state stability for discrete-time nonlinear systems,” *Automatica*, vol. 37, pp. 857–869, 2001.
- [247] C. F. Caruntu, A. E. Balau, M. Lazar, P. P. J. van den Bosch, and S. Di Cairano, “A predictive control solution for driveline oscillations damping,” in *Hybrid Systems: Computation and Control*, Chicago, USA, 2011, pp. 181–190.
- [248] C. F. Caruntu and C. Lazar, “Robust MPC for TrueTime simulation of a vehicle drivetrain controlled through CAN,” in *16th IEEE International Conference on Emerging Technologies and Factory Automation*, Toulouse, France, 2011.
- [249] A. Badawy, J. Zuraski, F. Bolourchi, and A. Chandy, “Modeling and analysis of an electric power steering system,” Tech. Rep. 01-0399, 1999.
- [250] A. Marouf, C. Sentouh, M. Djemai, and P. Pudlo, “Control of an electric power assisted steering system using reference model,” in *50th IEEE Conference on Decision*

and Control and European Control Conference, Orlando, FL, USA, 2011, pp. 6684–6690.

- [251] G. Naus, R. Vugts, J. Ploeg, R. V. Molengraft, and M. Steinbuch, “Towards On-The-Road Implementation Of Cooperative Adaptive Cruise Control,” in *16th ITS World Congress and Exhibition on Intelligent Transport Systems and Services*, Stockholm, Sweden, 2009.
- [252] V. Milanés, S. E. Shladover, J. Spring, C. Nowakowski, H. Kawazoe, and M. Nakamura, “Cooperative Adaptive Cruise Control in Real Traffic Situations,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, pp. 296–305, 2014.
- [253] R. Rajamani and C. Zhu, “Semi-autonomous Adaptive Cruise Control systems,” *IEEE Transactions on Vehicular Technology*, vol. 51, pp. 1186–1192, 2002.
- [254] A. Tiganasu, C. Lazar, and C. F. Caruntu, “Design and simulation evaluation of cooperative adaptive cruise control for a platoon of vehicles,” in *20th International Conference on System Theory, Control and Computing*, Sinaia, Romania, 2016.
- [255] K. J. Astrom and T. Hagglund, *Advanced PID Control*. ISA - Instrumentation, Systems, and Automation Society, USA, 2006.
- [256] E. F. Camacho and C. Bordons, *Model Predictive Control in the Process Industry*. Springer-Verlag, London, 1995.
- [257] A. Dutta, E. Hartley, J. Maciejowski, and R. De Keyser, “Certification of a class of industrial predictive controllers without terminal conditions,” in *53rd IEEE Conference on Decision and Control*, Los Angeles, USA, 2014, pp. 6695–6700.
- [258] M. Amling, “Low-cost implementation of vehicular platooning using pic microcontroller and diversified sensors,” in *The 2013 Research and Scholarship Symposium*, 2013.
- [259] A. A. Kheriji, F. Bouani, and M. Ksouri, “A microcontroller implementation of constrained model predictive control,” *International Journal of Electrical and Electronics Engineering*, vol. 5, 2011.
- [260] **A. Maxim**, C. M. Ionescu, C. F. Caruntu, C. Lazar, and R. De Keyser, “Reference tracking non-cooperative distributed mpc algorithm for multiple-agent systems: Application to a sextuple tank process,” *Journal of Process Control - under review*, 2017.
- [261] E. G. Hernandez-Martinez and E. Aranda-Bricaire, “Decentralized formation control of multi-agent robot systems based on formation graphs,” *Studies in Informatics and Control*, vol. 21, pp. 7–16, 2012.
- [262] E. G. Hernandez-Martinez, J. J. Flores-Godoy, and G. Fernandez-Anaya, “Decentralized discrete-time formation control for multirobot systems,” *Discrete Dynamics in Nature and Society*, 2013.

- [263] T. Arai, E. Pagello, and L. E. Parker, “Guest editorial: Advances in multirobot systems,” *IEEE Transactions on Robotics and Automation*, vol. 18, pp. 655–661, 2002.
- [264] K.-K. Oh, M.-C. Park, and H.-S. Ahn, “A survey of multi-agent formation control,” *Automatica*, vol. 53, pp. 424–440, 2015.
- [265] G. Antonelli, F. Arrichiello, F. Caccavle, and A. Marino, “Decentralized time-varying formation control of multi-robot systems,” *International Journal of Robotics Research*, vol. 33, pp. 1–15, 2014.
- [266] J. Yu and S. M. LaValle, “Distance optimal formation control on graphs with a tight convergence time guarantee,” in *IEEE Conference on Decision and Control*, Maui, Hawaii, USA, 2012, pp. 4023–4028.
- [267] A. Chevalier, C. Copot, S. Cristescu, C. Ionescu, and R. De Keyser, “Emulation of a highway bottleneck using leader-follower formation control,” in *IEEE 8th International Symposium on Applied Computational Intelligence and Informatics*, Timisoara, Romania, 2013.
- [268] A. Lopez-Gonzalez, E. D. Ferreira, E. G. Hernandez-Martinez, J. J. Flores-Godoy, G. Fernandez-Anaya, and P. Paniagua-Contro, “Multi-robot formation control using distance and orientation,” *Advanced Robotics*, vol. 30, pp. 901–913, 2016.
- [269] X. Huppe, J. de Lafontaine, M. Beauregard, and F. Michaud, “Guidance and control of a platoon of vehicles to changing environment conditions,” in *IEEE International Conference on Systems, Man and Cybernetics*, Washington, DC, USA, 2003.
- [270] N. Neuedorf and T. Bruns, “The vehicle platoon controller in the decentralised, autonomous intersection management of vehicles,” in *IEEE International Conference on Mechatronics*, Istanbul, Turkey, 2004, pp. 375–380.
- [271] S. Mastellone, D. M. Stipanovic, C. R. Graunke, K. A. Intlekofer, and M. W. Spong, “Formation control and collision avoidance for multi-agent nonholonomic systems: Theory and experiments,” *International Journal of Robotics Research*, vol. 27, pp. 107–126, 2008.
- [272] H. Diab, M. G. C. Grunewald, I. B. Makhlof, D. Abel, and S. Kowalewski, “A testing platform for cooperative vehicle platoon controllers,” in *13th International IEEE Annual Conference on Intelligent Transportation Systems*, Madeira Island, Portugal, 2010, pp. 1718–1723.
- [273] B. D. Anderson, Z. Lin, and M. Deghat, “Combining distance-based formation shape control with formation translation,” in *Developments in Control Theory Towards Global Control*, L. Qiu, J. Chen, T. Iwasaki, and H. Fujioka, Eds. IET, 2012, ch. 13, pp. 121–130.
- [274] A. El-Sayed Bayoumi Ibrahim, “Decentralized non-linear control of leader-follower formation of multiple autonomous mobile robots,” *International Journal of Applied Engineering Research*, vol. 11, pp. 6583–6590, 2016.

- [275] M. H. Yamchi and R. M. Esfanjani, “A decentralized receding horizon optimal approach to formation control of networked mobile robots,” *Optimal Control Applications and Methods*, 2017.
- [276] D. V. Neamtu, E. Fabregas, B. Wyns, R. De Keyser, S. Dormido, and C. M. Ionescu, “A remote laboratory for mobile robot applications,” in *18th IFAC World Congress*, Milano, Italy, 2011, pp. 7280–7285.
- [277] E. F. Camacho and C. Bordons, *Model Predictive Control*. Springer Verlag, 2004.
- [278] R. De Keyser, *Model based Predictive Control for Linear Systems*. Oxford: Eolss Publishers Co Ltd, 2003, vol. UNESCO Encyclopaedia of Life Support Systems <http://www.eolss.net>, Article contribution 6.43.16.1, 35 pages.
- [279] B. Kocaturk, “Motion control of wheeled mobile robots,” *Interdisciplinary Description of Complex Systems*, vol. 13, pp. 41–47, 2015.
- [280] D. Swaroop, J. K. Hedrick, C. C. Chien, and P. Ioannou, “A comparison of spacing and headway control laws for automatically controlled vehicles,” *Vehicle System Dynamics: International Journal of Vehicle Mechanics and Mobility*, vol. 23, pp. 597–625, 1994.
- [281] J. P. Hespanha, *Linear systems theory*. Princeton University Press, 2009.
- [282] J. Eyre, D. Yanakiev, and I. Kanellakopoulos, “A simplified framework for string stability analysis of automated vehicles,” *Vehicle System Dynamics: International Journal of Vehicle Mechanics and Mobility*, vol. 30, pp. 375–405, 1998.
- [283] L. C. Westphal, *Sourcebook of Control Systems Engineering*. Springer US, 1995.
- [284] J. Harding, G. R. Powell, R. Yoon, J. Fikentscher, C. Doylec, D. Sade, M. Lukuc, J. Simons, and J. Wang, “Vehicle-to-vehicle communications: Readiness of V2V technology for application,” National Highway Traffic Safety Administration, Washington, DC, Tech. Rep., 08 2014.
- [285] W. Chen, *Vehicular Communications and Networks*. Woodhead Publishing, 2015.
- [286] L. Zha, P. Songchitruksa, and K. N. Balke, “Next generation safety performance monitoring at signalized intersections using connected vehicle technology,” Texas A&M Transportation Institute, Tech. Rep., 08 2014.
- [287] O. Gietelink, J. Ploeg, B. De Schutter, and M. Verhaegen, “Development of advanced driver assistance systems with vehicle hardware-in-the-loop simulations,” *Vehicle System Dynamics*, vol. 44, pp. 569–590, 2006.
- [288] C. Phillips, M. Ton, D. Sicker, and D. Grunwald, “Practical radio environment mapping with geostatistics,” in *IEEE International Symposium on Dynamic Access Networks*, Bellevue, WA, USA, 2012, pp. 422–433.
- [289] *mOways User Manual*, 2012.

- [290] *PIC18F87J50 Family Data Sheet*. Microchip Technology, 2009.
- [291] A. Tanenbaum, *Computer Networks*. Prentice Hall PTR, 2003.
- [292] C. Copot, C. F. Caruntu, J. P. S. De Azevedo, and R. De Keyser, “Networked communications over a lab-scale test bench for autonomous highway systems,” in *19th International Conference on System Theory, Control and Computing*, Cheile Gradistei, Romania, 2015, pp. 454–459.