

LPV alapú irányítástervezés automatizált járművek számára adatvezérelt módszerek alkalmazásával

Fényes Dániel* Németh Balázs**

*Budapesti Műszaki és Gazdaságtudományi Egyetem,
Közlekedés- és Járműirányítási Tanszék
H-1111 Budapest, Műgyetem rkp. 3,
e-mail: feny.es.daniel@mail.bme.hu

**Számítástechnikai és Automatizálási Kutatóintézet,
Rendszer és Irányításméleti Kutatólaboratórium
H-1111 Budapest, Kende u. 13-17,
e-mail:nemeth.balazs@sztaki.hu

Absztrakt: Jelen cikkben egy újszerű adatvezérelt modellezési megközelítés és irányítástervezés kerül bemutatásra autonóm járművek számára. Az irányítás-orientált politópikus modell gépi tanulási algoritmus felhasználásával kerül meghatározásra lineárisan változó paraméterű (LPV) struktúrában. Az irányítási algoritmus az LPV alapú szintézisen nyugszik, amely képes garantálni a kívánt minőségi jellemzőket. Továbbá az oldalirányú szabályozás mellett, a jármű hosszirányú sebessége is módosításra kerül egy optimalizálási algoritmuson keresztül. A javasolt irányítási rendszer működése és hatékonyság egy átfogó szimulációs példán keresztül kerül bemutatásra egy nagy pontosságú szimulációs szoftver, a CarSim felhasználásával.

1. BEVEZETÉS

Manapság az autópia egyik fő kihívása az önvezető járművek fejlesztése. Ehhez a kihíváshoz új technológiákra és megoldásokra van szükség, amelyek garantálhatják a jármű biztonságos és stabil mozgását. Ezen oknál fogva szükséges a különböző területek együttműködése, mint például az érzékelés (jármű és környezete), a kommunikáció (V2V és V2X), a döntéshozatal és a beavatkozások. A cikk az utolsó említett területhez kapcsolódóan bemutat egy új adaptív, adatvezérelt irányítási struktúrát autonóm járművek szabályozásához. Az elmúlt évek során számos módszert fejlesztettek ki járművek irányítására, amelyek többsége egy fizikai modellre épül. Annak ellenére, hogy a használt modellek egy része rendkívül összetett, nem tudják pontosan modellezni a jármű nemlineáris dinamikáját, sőt, megnehezítik a vezérlés kialakítását, lásd (Németh *et al.*, 2016, Masouleh *et al.*, 2016). Egy másik lehetséges megközelítés lehet a mélytanulás és a gépi tanuláson alapuló irányítási rendszerek, Pl. in (Hubschneider *et al.*, 2017, Rausch *et al.*, 2017). Az említett neurális háló alapú megoldások általában képesek kezelni a jármű nemlineáris viselkedését és a megfelelő performanciákat kielégíteni, egy jelentős hátrányuk, hogy a stabilitásuk nehezen vizsgálható. Emiatt a közúti járművekben történő alkalmazása komoly kockázatot hordoz magával.

Jelen cikkben bemutatásra kerülő adatvezérelt irányítási algoritmus egyik nagy előnye, hogy a stabilitása a klasszikus módszerek alkalmazásával meghatározható és biztosítható. Az irányítási rendszer felépítése az LPV (Linear Parameter Varying) módszeren alapszik, amelynek az ütemezési változói gépi tanulási módszerrel kerülnek

meghatározásra. E megközelítések kombinálásával a jármű nemlineáris viselkedése figyelembe vehető, és ezzel párhuzamosan, a rendszer stabilitása és az előírt performanciák biztosíthatók. Hasonló megközelítések találhatók a korszerű szakirodalomban is: A (Borrelli *et al.*, 2018) cikkben egy MPC (Model Predictive Control) alapú megoldást mutatnak be, amelynek terminálkölségét és terminálkészleteit iteratív megközelítéssel határozzák meg. Egy másik példa a (Fliess *et al.*, 2013), amely egy MFC-t (Model Free Control) megközelítést ismertet. Az MFC módszer legfőbb előnye, hogy nem igényel előzetes ismereteket a rendszer felépítéséről, és képes kezelni nagymértékű nemlinearitással rendelkező rendszereket is. A módszer legfőbb hátránya, hogy a megfelelő performanciák eléréséhez nincs kidolgozott módszertan, így iteratív lépéseken keresztül érhetőek el a kívánt tulajdonságok.

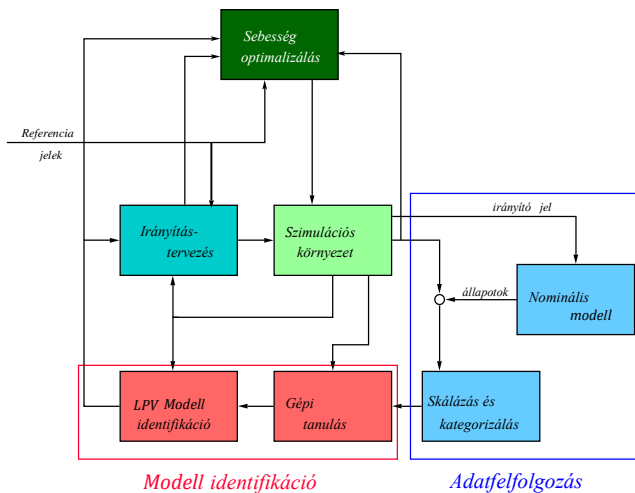
A bemutatásra kerülő irányítási algoritmus egy nominális fizikai modellre épít, amelynek az egyes paramétereit egy optimalizálási eljárás során kerülnek meghatározásra specifikus munkapontok környezetében. Továbbá a munkapontként meghatározott modellek egy politópikus rendszert alkotnak, amely így LPV modellosztály módszereivel kezelhető. A kapott politópikus modell ütemezési változói egy gépi tanulási módszerrel kerülnek meghatározásra felhasználva a járműmozgásából képzett nagy adathalmazokat. Fontos megjegyezni, hogy a javasolt módszer független az adott irányítási problémától és a rendszer dinamikai tulajdonságaitól. Ebben a cikkben a probléma az automatizált vezetés kontextusában kerül bemutatásra. A irányítási rendszernek két fő rétege van. Az első réteg feladata a jármű

oldalirányú mozgásának irányítása, így a pályakövetés garantálása is. A második réteg a hosszanti sebesség maximalizálási módszerét mutatja be. Az optimalizálás során a sebességet úgy választják meg, hogy a jármű prediktált mozgását egy biztonsági tartományon belül tartsa. A javasolt módszer működése és hatékonysága egy komplex scenárión keresztül kerül bemutatásra, amely egy nagypontosságú szimulációs szoftverben, a CarSim-ben kerül szimulálásra.

1.1 Az irányítási rendszer felépítése

A következőkben bemutatásra kerül a modellezés és az irányítástervezés lépései.

1. ábra illusztrálja az irányítási rendszer felépítését. A több különböző aleggységéből épül fel úgy, mint: Adatfeldolgozás, Modell identifikáció, Irányítástervezés, Sebesség optimalizálás, Szimulációs környezet.



1. ábra: Irányítási rendszer struktúrája

A „Szimulációs környezet” magában foglalja a kutatásban használt nagy pontosságú CarSim szimulációs szoftvert. Ennek a szoftvernek két fő szerepe van: 1. A gépi tanulási algoritmus adatkészleteinek előállítására. 2. Az irányítási rendszer működőképességének a demonstrálása.

Az „Adatfeldolgozás” réteg a kapott adatok előzetes feldolgozását végzi úgy, mint skálázás és kategorizálás.

A „Modell identifikáció” magában foglalja az ütemezési paraméterek kiszámítását. A nagy adathalmazok felhasználásával meghatározza azokat a változókat, amelyek a legnagyobb hatással vannak a modell pontosságára. Továbbá ebben a rétegben kerül kiszámításra a politópikus modell egyes paramétereinek az értékei.

Az „Irányítástervezés” a jármű oldalirányú szabályozójának a tervezési lépéseit rejti magában.

A „Sebesség optimalizálás” réteg célja a jármű számára az optimális hosszirányú sebesség kiszámítása, amely garantálja a jármű biztonságos és stabil mozgását.

Végül egy átfogó szimuláció kerül prezentálásra a javasolt irányítási rendszer működésének és hatékonyságának bemutatására az 5. fejezetben.

2. LPV RENDSZER ADATVEZÉREL T IDENTIFIKÁCIÓJA

Ebben a részben az LPV alapú modellezési folyamatot mutatjuk be. Először is, az adathalmazok felvételének a módszertana kerül ismertetésre, amelyeket a gépi tanulási algoritmus használ fel.

2.1 Adathalmazok generálása

Adathalmazok generálása a már említett CarSim szimulációs szoftver felhasználásával történik. A szoftverben számos szimuláció került végrehajtásra változó paraméterek alkalmazásával annak érdekében, hogy a jármű működési tartományának minél szélesebb köre lefedésre kerüljön. A szimulációk során a járművet a szoftver beépített vezetői modellje vezette különböző hosszirányú sebességek mellett: $v_x=10-20$ m/s. A scenáriók során a következő jellemzők kerültek rögzítésre és lementésre:

1. Hosszirányú sebesség,
2. Kerekek szögsebességei,
3. Legyezési szögsebesség,
4. Kormányzási szög,
5. Gyorsulások,
6. Oldalkúszási szög.

A jármű oldalkúszási szöge kizárólag a modellezési eljárásban kerül felhasználásra, az irányító rendszer működéséhez nem szükséges.

2.2 Az összegyűjtött esetek kategorizálása

Második lépésként az összegyűjtött példányok kategóriákba lettek rendezve, amelyek tükrözik a jármű nemlineáris viselkedését. Ezen kategorizálás úgy történik, hogy az egyes példányok eltérése a nominális fizikai modelltől kiszámításra kerül. A fizikai modell alapvetően két egyenletről áll, lásd (Rajamani *et al.*, 2005):

$$I\ddot{\psi} = C_1\alpha_1l_1 - C_2\alpha_2l_2 \quad (1)$$

$$mv_x(\dot{\psi} + \dot{\beta}) = C_1\alpha_1 + C_2\alpha_2 \quad (2)$$

$\alpha_{1,2}$ az egyes kerekek oldalkúszási szögei, $\dot{\psi}$ a legyezési szögsebesség, β jelenti az oldalkúszási szöveget, m a jármű tömege I a legyezési inercia, $C_{1,2}$ kanyarodási merevségek és $l_{1,2}$ a jármű geometriai paraméterei.

A bemutatott járműmodell a következő állapotter reprezentációba kerül átírásra: $\dot{x} = Ax + bu$, ahol az állapotvektor: $x = [\dot{\psi}, \beta]$. Továbbá a folytonos állapotter reprezentáció diszkrétizálásra kerül $T_s = 0.01s$ mintavételi idő alkalmazásával.

Az felvett adathalmaz által tartalmazott egyes esetek a mért és diszkrétizált modell által számított állapotok különbségként kerülnek kategorizálásra.

$$\dot{\psi}_e(t_i) = \frac{|\dot{\psi}_m(t_i) - \dot{\psi}_n(t_i)|}{\dot{\psi}_n(t_i)} \quad (3)$$

$$\beta_e(t_i) = \frac{|\beta_m(t_i) - \beta_n(t_i)|}{\beta_n(t_i)} \quad (4)$$

ahol $\dot{\psi}_n$ és β_n a diskretizált modell által számított állapotok, amíg $\dot{\psi}_m$ és β_m a mért állapotokat jelölik a t_i -ik időpillanatban. A kapott hibafüggvények $\dot{\psi}_e$ és β_e előre definiált ekvidisztánsan felosztott 0 és 1 közötti kategóriákba kerülnek besorolásra a következő módon:

$$cat(t_i) = \max \left(f(\dot{\psi}_e(t_i)), f(\beta_e(t_i)) \right), \quad (5)$$

ami azt jelenti, hogy a $cat(t_i)$ kategóriát a hibák maximuma határozza meg.

2.3 Az ütemezési változók kiválasztása

Az előző alfejezetben az egyes esetek előre definiált kategóriákba ($cat(t_i)$) kerültek besorolásra. A kategorizálás az egyes esetek névleges modelltől való eltérésétől függ, amelyet offline módon, az összegyűjtött adatok felhasználásával lehet számítani. A jármű működése közben azonban ez nem lehetséges, emiatt egy döntési fa algoritmus segítségével kerülnek osztályozásra az egyes esetek a jármű működése közben. A döntési fa algoritmusok képesek leképezni a nagymértékben nemlineáris viselkedést is, így megtalálni a kapcsolatot a bemeneti változók (mért jelek) és kimenet között (kategória). Ebben a cikkben a jól ismert döntési fa algoritmus, C4.5 kerül alkalmazásra. A következőkben az algoritmus működése kerül bemutatásra.

Minden döntési fa három fő elemből áll: levelekből, ágakból és csomópontokból. A csomópontok feltételek vannak hozzárendelve, pl. egy változó értéke kisebb / nagyobb, mint egy adott szám. Egy csomópontnak legalább két eredménye van (attól függően, hogy a feltétel teljesült-e vagy sem). Ezek az eredmények a fa ágai, amelyek egy másik csomóponthoz vagy levélhez kapcsolódhatnak. Ebben az esetben a levelek egy-egy adott kategóriát jelentenek. Első lépésként az algoritmus két részhalmazra osztja az adatkészletet: tanulási és tesztkészletre. Az tanulókészlet a fa felépítésére, míg a tesztkészlet az előállított fa értékelésére szolgál. Ezen algoritmusok alapjában a halmazok entrópiájának minimalizálása új részhalmazok (új feltételek, csomópontok) létrehozásával. Részletesebb leírás megtalálható (Hunt *et al.*, 1962, Quinlan *et al.*, 1993) cikkekben. Összefoglalva, a döntési fa meghatározza az esetekhez tartozó kategóriákat (ξ), amelyek az LPV rendszerben ütemezési paraméterként szolgálnak.

Minden kategóriához meg kell határozni egy lineáris rendszert, amelyből felépíthető a politópikus LPV rendszer. Ahogy már említésre került, az LPV rendszer megőrzi az eredeti, nominális rendszer szerkezetét, így a lineáris rendszereknek két állapota van: $x_d = [\dot{\psi} \ \beta]$. Az állapotér reprezentáció így a következőképp adható meg:

$$A_d(\xi) = \begin{bmatrix} a_{11}(\xi) & a_{12}(\xi) \\ a_{21}(\xi) & a_{22}(\xi) \end{bmatrix}, \quad B_d(\xi) = \begin{bmatrix} b_1(\xi) \\ b_2(\xi) \end{bmatrix} \quad (6)$$

ahol $a_{ij}(\xi), b_i(\xi)$ paraméterek, amelyeket meg kell határozni minden egyes kategóriához (ξ). A rendszer bemenete a kormánysszög $u_d = \delta$.

Az identifikációs eljárás célja a mért állapotok ($x_{m,j}(t_i)$) és a kiszámított állapotok ($x_{n,j}(t_i)$) közötti hibák minimalizálása minden egyes lépésnél. Az identifikációs probléma a következő optimalizálási feladatként írható fel:

$$\min_{\substack{a_{11}(\xi), a_{12}(\xi), a_{21}(\xi), \\ a_{22}(\xi), b_1(\xi), b_2(\xi)}} \sum_{i=1}^N \sum_{j=1}^2 (x_{m,j}(t_i) - x_{n,j}(t_i))^2 \quad (7)$$

A kapott másodfokú optimalizálási problémát meg kell oldani az összes rögzített ξ kategóriára, lásd (Gill *et al.*, 1981, Coleman *et al.*, 1996), így a származtatott lineáris rendszerekből felépíthető a kívánt LPV rendszer.

3. LPV-ALAPÚ LATERÁLIS IRÁNYÍTÁS TERVEZÉS

Az LPV-alapú irányítástervezés fő célja a jármű pályakövetésének a biztosítása. Emiatt szükséges a meghatározott lineáris modellek kibővítése az oldalirányú pozícióval, mint állapottal, amely a $v_x(\dot{\psi} + \beta) = \dot{y} + \dot{\psi}v_x$ összefüggés alapján elvégezhető. A rendszer kibővített állapotvektora $x_d = [\dot{\psi} \ \beta \ \dot{y} \ y]$ és az ütemezési változók: $\rho = [v_x \ \xi]$.

Az irányítási algoritmusnak teljesítenie kell bizonyos minőségi kritériumokat, amelyek a következőképpen formalizálhatóak:

- Az irányítási algoritmus fő célja a jármű oldalirányú pozíciójának a szabályozása, és így a jármű pályakövetésének a biztosítása:

$$z_1 = y_{ref} - y, |z_1| \rightarrow \min, \quad (8)$$

- Másik fontos követelmény a jármű legyezési szögsebességének a szabályozása, amely által biztosítható a sima útkövetés:

$$z_2 = \dot{\psi}_{ref} - \dot{\psi}, |z_2| \rightarrow \min, \quad (9)$$

- Természetesen a beavatkozó jel minimalizálása is nagy jelentőséggel bír az irányító rendszer működése során:

$$z_3 = \delta, |z_3| \rightarrow \min. \quad (10)$$

A bemutatott követelmények a következő vektorban foglalhatóak össze: $z = [z_1 \ z_2 \ z_3]^T$ amely a következő performancia egyenlethez vezet:

$$z = C_1 x + D_{11} r + D_{12} u, \quad (11)$$

ahol C_1, D_{11}, D_{12} mátrixok, és r tartalmazza a y_{ref} jelet.

Végül az eredeti állapotegyenleteket kibővítjük a performancia egyenletekkel:

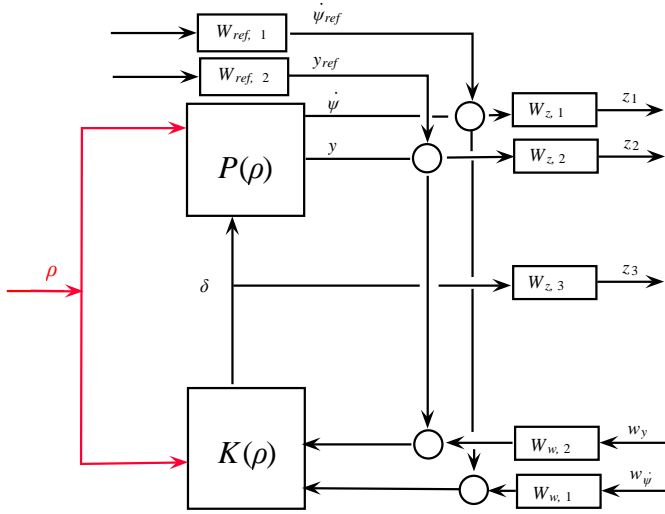
$$\dot{x} = Ax + Bu, \quad (12)$$

$$z = C_1x + D_{11}r + D_{12}u, \quad (13)$$

$$y_K = C_2x, \quad (14)$$

$$y_K = [y \ \dot{\psi}] \quad (15)$$

A bemutatott minőségi kritériumok biztosításához különböző súlyozó függvények kerültek meghatározásra. $W_{ref,1}$ és $W_{ref,2}$ a referenciajelek skálázási függvényei. $W_{z,1}$, $W_{z,2}$ és $W_{z,3}$ az előírt kritériumok biztosítására szolgáló súlyozási függvények. $W_{w,1}$ és $W_{w,2}$, a zajok csillapítására szolgáló függvények.



2. ábra: Irányítási rendszer struktúrája

Végül a kívánt LPV szabályzó a következő L_2 norma (a zavarás és a minőségi kritériumok között) minimalizálásval kapható meg:

$$\inf_{K(\rho)} \sup_{\rho \in F_\rho} \sup_{\|w\|_2 \neq 0, w \in \mathcal{L}_2} \frac{\|z\|_2}{\|w\|_2} \quad (16)$$

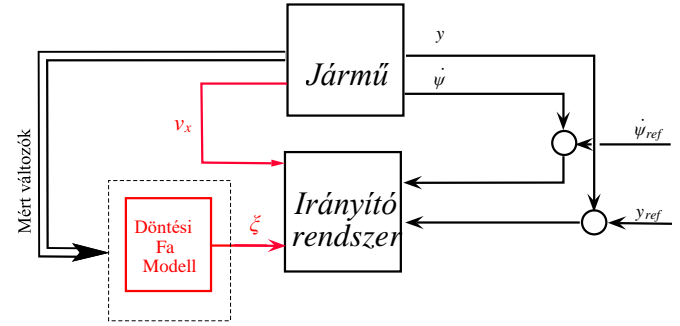
ahol F_ρ korlátozza az ütemezési változókat.

A teljes rendszer felépítését a 3. ábra szemlélteti.

4. SEBESSÉGPROFIL OPTIMALIZÁLÁSA

A sebességprofil megfelelő megválasztása jelentősen hozzájárul a jármű dinamikai tulajdonságaihoz és stabilitásához. Például, amikor a jármű nagy sebességgel éles kanyart tesz meg, az oldalirányú erők könnyen elérhetik csúcserőértékeket, ami a jármű irányíthatóságának elvesztését eredményezheti. Továbbá a jármű szükségtelen lassú mozgása is negatív hatással lehet a forgalomra. Ezért fontos a jármű sebességének maximalizálása oly módon, hogy a jármű stabilitása fennmaradjon. Így ezen probléma egy

optimalizálási feladathoz vezet, amely a fejezet keretein belül kerül ismertetésre.



3. ábra: Irányítási rendszer struktúrája

Az említett probléma egy lehetséges megoldását mutatja be (Palmieri, *et al.*, 2012), ahol a jármű stabil régióinak a figyelembevételével kerül kiszámításra az optimális sebesség. Az cikk a jármű stabil régióit egy fizikai modell segítségével határozza meg. A koncepció lényegében a darabonkénti lineáris modell azon területeit határozza meg az állapot-terben, ahol elkerülhető az oldalirányú dinamika túlírányítása és nemlineáris jellemzői. Ebben a cikkben egy új meghatározási módszer kerül bemutatásra, amely a stabil régiókat az adatvezérelt politópikus LPV modell segítségével számítja ki. Ezzel a módszerrel a stabil régiók pontosabban számíthatóak ki.

Az optimális sebesség profil kiválasztásához a vezérelt jármű oldalirányú mozgáspredikciója szükséges. A zárt hurkú rendszer az adatvezérelt LPV modellek és a bemutatott vezérlő segítségével számítható, mint:

$$A_{cl}(\rho_1) = \begin{bmatrix} A_{cl,11} & A_{cl,12} \\ A_{cl,21} & A_{cl,22} \end{bmatrix}, \quad (17)$$

$$A_{cl,11} = A(\rho) + B(\rho)D_K(\rho)C_2$$

$$A_{cl,12} = B(\rho)C_K(\rho),$$

$$A_{cl}(\rho_1) = \begin{bmatrix} A_{cl,11} & A_{cl,12} \\ A_{cl,21} & A_{cl,22} \end{bmatrix}, \quad (18)$$

$$A_{cl,11} = A(\rho) + B(\rho)D_K(\rho)C_2$$

$$A_{cl,12} = B(\rho)C_K(\rho),$$

A zárt hurkú rendszer predikciójához szükséges mind az identifikált mind az LPV szabályzó diszkrétizálása $T_s = 0.01s$ mintavételi idő alkalmazásával, lásd (Tóth, 2010). A jármű mozgása a zárt hurkú rendszer alapján számítható az n hosszúságú predikciós horizonton.

Az optimalizálási feladat végső célja a jármű sebességének a maximalizálása az említett n hosszúságú horizonton garantálva a következő kényszer betartását:

$$\xi |y_{cl}(k, n) \leq \xi_{max}, \quad \forall k \leq i \leq n. \quad (19)$$

ahol $\xi |y_{cl}(k, n)$ a prediktált állapotokhoz tartozó kategóriát.

Végül a hosszirányú sebesség optimalizálási feladat formalizálható, mint:

$$\max_{v_{x(k+1)} \dots v_{x(k+n)}} \sum_{i=k}^n v_x(i) \quad (20)$$

s.t.

$$\xi |y_{cl}(k, n)| \leq \xi_{max}, \quad \forall k \leq i \leq n \quad (21)$$

Az optimalizálási eljárás eredménye egy v_x sebességprofil, amely garantálja a jármű stabilitását n hosszúságú horizont.

5. SZIMULÁCIÓS EREDMÉNYEK

Továbbiakban egy szimulációs példa kerül bemutatásra, amely illusztrálja a javasolt irányítási algoritmus hatékonyságát és működőképességét.

Az LPV modellezés során használt kategóriák száma hatra lett választva a szimulációk során. Ezen felosztást alkalmazva több döntési fa lett létrehozva, amelyek jellemző értékeit az alábbi táblázat tartalmazza. A táblázat oszlopai mutatják a létrehozás során használt jeleket, a levelenként előforduló esetek minimális számát, a helyesen osztályozott esetek százalékos arányát és a fa méretét (elemeit). A „levelenkénti minimális esetszám” a fa tervezési paramétere, amellyel a fa mérete befolyásolható. Mint látható, szoros összefüggés van a fa mérete és pontossága között. A gyakorlatban a nagyméretű fák nem alkalmazhatók, ezért egyensúlyt kell találni a pontossága és a mérete között. Ebben az esetben a legpontosabb fának is csak 51 eleme van, ezért ez kerül felhasználásra a következőkben.

Jelek	Min. obj.	Hely. Esetek	Méret
δ, ψ, a_y, v_x	500	96.37%	51
δ, ψ, v_x	1000	95.82%	33
δ, ψ	2000	94.54%	19
δ, a_y	3000	93.21%	17
δ	5000	91.54%	13

1. táblázat: Generált döntési fák

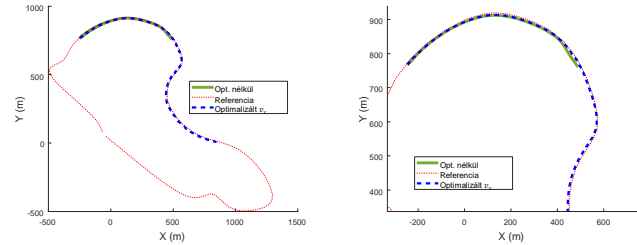
A második táblázat a kiválasztott fa konfúziós mátrixát mutatja. A konfúziós mátrix a helyesen és rosszul osztályozott esetek százalékos arányát mutatja meg.

Amint azt a táblázat is szemlélteti, a téves osztályozás százalékos aránya alacsony, és a tévesen besorolt összes eset maximálisan egy kategóriára helyezkedik el a helyes osztálytól. Így, ezen döntési fa felhasználható az irányító rendszer tervezése során.

$\xi = 0$	$\xi = 1$	$\xi = 2$	$\xi = 3$	$\xi = 4$	$\xi = 5$	
57.86	0.7	0	0	0	0	cat = 0
0.6	27.2	0.5	0	0	0	cat = 1
0	0.5	8.3	0.5	0	0	cat = 2
0	0	0.24	3.87	0.24	0	cat = 3
0	0	0	0.2	1.3	0	cat = 4
0	0	0	0	0.24	1.54	cat = 5

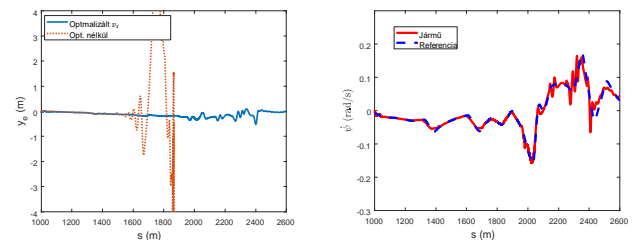
2. táblázat: Kiválasztott döntési fa konfúziós mátrixa

A szimulációk során a járművet kétszer halad végig a Melbourne Grand Prix pálya egy szakaszán. Az első esetben a személygépkocsit egy bemutatott irányítórendszer vezérli a sebességoptimalizálási folyamat segítségével. Míg a második szimulációban a járművet a névleges sebességprofil halad végig a pályán.



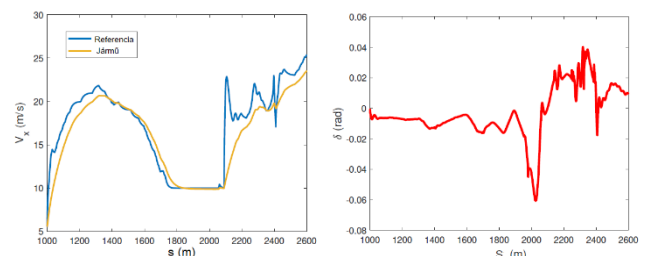
4. ábra: Jármű által bejárt pályaszakasz

A jármű által bejárt útszakaszt a 4. ábra mutatja mindkét esetre nézve. Ahogy az ábrák is mutatják, az előre meghatározott sebességprofil követő jármű nem képes követni a pályát, az első kanyarban elhagyja azt, amíg a másik esetben az optimalizált sebességprofittal rendelkező jármű kis oldalirányú hibával képes végighaladni a kijelölt útszakaszon. A jobb oldali ábrán a legyezési szögsebesség követése látható. Amint azt az ábra is szemlélteti, a javasolt rendszer az oldalirányú pozícióhoz hasonlóan képes garantálni az oldalirányú szögsebesség követését is.



5. ábra: Jármű oldalirányú hibája és legyezési szögsebessége

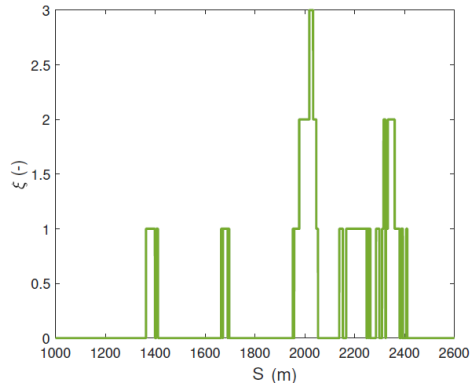
Az oldalirányú hibát külön illusztrálja az 5. ábra. Látható, hogy a jármű oldalirányú hibája az optimalizált sebességprofilt használva lényegesen kisebb, maximuma körülbelül 0,4m. Ezzel szemben a másik esetben a jármű nagy oldalirányú hibával rendelkezik, így el is hagyja a kijelölt utat. A 6. ábra a kiszámított sebesség profilt, mint referencia sebességet és a jármű által megvalósított profilt mutatja. A másik ábrán a kormányzószög látható.



6. ábra: Jármű sebessége és kormányzószöge

Ahogy az ábra mutatja, a kormányzószög [-0.06rad - 0.04rad] között változik, ami megfelelő tartomány egy szokványos

személygépköcsi kormányrendszerét tekintve. Az ütemezési változó értékét döntési fa számítja ki, amely a δ, ψ, a_y, v_x változókat használja. Látható, hogy ennek a paraméternek a maximuma $\xi = 3$, ami azt jelenti, hogy a jármű a szimuláció során nem lépi túl a felső határt, így a jármű stabilitása az út teljes hosszán garantálásra került.



7. ábra: Ütemezési változó alakulása

6. KONKLÚZIÓ

A cikkben bemutatásra került egy új, adatvezérelt rendszer identifikációs módszer, amely felhasználásra is került egy személygépjármű szabályozási problémára.

Ennek a megközelítésnek az a fő előnye, hogy az elvárt minőségi kritériumok széles működési tartományban garantálhatóak, amely a bemutatott identifikációs módszernek köszönhető. Továbbá, bemutatásra került egy sebesség optimalizálási eljárás, amely képes garantálni, hogy a jármű elkerülje a nemlineáris mozgási tartományát. A továbbfejlesztett LPV alapú irányítási rendszer és a sebesség optimalizálási módszer együttműködésével garantálható a jármű stabilitása és pályakövetése szélsőséges esetekben is.

KÖSZÖNETNYILVÁNÍTÁS

EFOP-3.6.3-VEKOP-16-2017-00001: Tehetséggondozás és kutatói utánpótlás fejlesztése autonóm járműirányítási technológiák területén - A projekt a Magyar Állam és az Európai Unió támogatásával, az Európai Szociális Alap társfinanszírozásával valósul meg.

Fényes Dániel kutatása az Innovációs és Technológiai Minisztérium ÚNKP-20-3 kódszámú Új Nemzeti Kiválósági Programjának a Nemzeti Kutatási, Fejlesztési és Innovációs Alapból finanszírozott szakmai támogatásával készült.

Németh Balázs kutatása a Bolyai János Kutatási Ösztöndíj és az Innovációs és Technológiai Minisztérium ÚNKP-20-5 kódszámú Új Nemzeti Kiválósági Programjának a Nemzeti Kutatási, Fejlesztési és Innovációs Alapból finanszírozott szakmai támogatásával készült.

IRODALOMJEGYZÉK

T. F. Coleman and Y. Li. A reflective newton method for minimizing a quadratic function subject to bounds on some of the variables. *SIAM Journal on Optimization*, 6(4):1040–1058, 1996.

Michel Fliess and Cédric Join. Model-free control. *International Journal of Control*, 86(12):2228–2252, 2013.

P. E. Gill, W. Murray, and M.H. Wright. *Practical Optimization*. Academic Press, London UK, 1981.

C. Hubschneider, A. Bauer, J. Doll, M. Weber, S. Klemm, F. Kuhnt, and J. M. Zöllner. Integrating end-to-end learned steering into probabilistic autonomous driving. In *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*, pages 1–7, Oct 2017.

E. B. Hunt. *Concept Learning: An information Processing Problem*. Wiley, 1962.

M. I. Masouleh and D. J. N. Limebeer. Non-linear vehicle domain of attraction analysis using sum-of-squares programming. In *2016 IEEE Conference on Control Applications (CCA)*, pages 1209–1214, Sept 2016.

B. Németh, P. Gáspár, and T. Péni. Nonlinear analysis of vehicle control actuations based on controlled invariant sets. *Int. J. Applied Mathematics and Computer Science*, 26(1), 2016.

Giovanni Palmieri, Miroslav Baric, Luigi Glielmo, and Francesco Borrelli. Robust vehicle lateral stabilisation via set-based methods for uncertain piecewise affine systems. *Vehicle System Dynamics*, 50(6):861–882, 2012. J. R. Quinlan. C4.5: Programs for Machine Learning. Morgan Kaufmann Publishers, San Mateo, California, 1993.

R. Rajamani. *Vehicle dynamics and control*. Springer, 2005.

V. Rausch, A. Hansen, E. Solowjow, C. Liu, E. Kreuzer, and J. K. Hedrick. Learning a deep neural net policy for end-to-end control of autonomous vehicles. In *2017 American Control Conference (ACC)*, pages 4914–4919, May 2017.

U. Rosolia and F. Borrelli. Learning model predictive control for iterative tasks. a data-driven control framework. *IEEE Transactions on Automatic Control*, 63(7):1883 – 1896, July 2018.

Roland Tóth. Modeling and Identification of Linear Parameter-Varying Systems, volume 403 of *Lecture Notes in Control and Information Sciences*. Springer, Berlin, Heidelberg, 2010.