



## AUTONOMINIO AUTOMOBILIO VALDYMO ALGORITMO FORMULAVIMAS IR TEORINĖ ANALIZĖ

Paulius SKAČKAUSKAS<sup>1</sup>, Vilius MEJERAS<sup>2</sup>, Edgar SOKOLOVSKIJ<sup>3</sup>

*Vilniaus Gedimino technikos universitetas, Vilnius, Lietuva*

*El. paštas: <sup>1</sup>paulius.skackauskas@vgtu.lt; <sup>2</sup>vilius.mejeras@vgtu.lt; <sup>3</sup>edgar.sokolovskij@vgtu.lt*

**Santrauka.** Tobulėjant autonominių transporto priemonių technologijoms, vienas didžiausių uždavinių – užtikrinti saugią ir efektyvią autonominių transporto priemonių integraciją į transporto sistemą. Tokių transporto priemonių saugumas priklauso nuo jų galimybės stabiliai ir tiksliai judėti pageidaujama trajektorija. Intelektinis valdymo algoritmas yra svarbiausia sistema, kontroliuojanti autonominės transporto priemonės judėjimą. Todėl skirtingais metodais paremtų racionalių valdymo algoritmų formulavimas bei jų analizė yra reikšmingas dėmuo siekiant, kad autonominės transporto priemonės taptų visaverčiais eismo dalyviais. Straipsnyje aprašomas teorinis intelektinio valdymo algoritmo prototipas, skirtas tiksliai pageidaujamos judėjimo trajektorijos sekimui užtikrinti skirtingomis judėjimo sąlygomis. Algoritmo prototipo veikimas paremtas dviejų transporto priemonių matematinėmis modelių tarpusavio lyginimu ir optimizavimo uždavinio sprendimu. Aprašomas valdymo algoritmo prototipas pateikiamas programinio paketo *MATLAB / Simulink* aplinkoje, atliekama algoritmo racionalaus veikimo analizė.

**Reikšminiai žodžiai:** autonominis automobilis, genetinis algoritmas, judėjimo trajektorijos sekimas, nuokrypis.

### Įvadas

Tai, kad pagrindinių technologijų, reikalingų transporto sistemai tapti modernesnei, veiksmingesnei ir patogesnei vartotojams, kūrimas bei diegimas yra vis aktyviau remiamas, daro poveikį intensyviai intelektinių transporto sistemų, pagalbinių vairavimo sistemų ir autonominių transporto priemonių technologiniam tobulėjimui. Vis dėlto dėl aiškaus teisinio reguliavimo nebuvimo ir įvairių sistemų technologinio netobulumo, autonominės transporto priemonės vis dar nėra visaverčiai eismo dalyviai (Anderson *et al.* 2016). Vienas pagrindinių kriterijų, nuo kurio priklauso sėkminga autonominių transporto priemonių integracija į transporto sistemą, yra judėjimo saugumas, t. y. galimybė tiksliai ir stabiliai judėti pageidaujama trajektorija (Kalra, Paddock 2016).

Januševičius (2003) nurodo, kad, norint suformuluoti racionalų valdymo dėsnį, kuris užtikrintų stabilų ir tikslų judėjimą, pirmiausia turi būti analizuojama dinaminė sistemos elgsena ir bendrieji judėjimo dėsniniai. Anot Januševičiaus (2003), jeigu valdoma sistema yra nestabili, nėra tikslinga analizuoti kitų jos valdymo statinių ir dinaminų charakteristikų, nes ji negali įvykdyti nustatytų valdymo tikslų. Nagrinėjant bet kokią kūnų sistemą, ji supranta kaip nestabili, jeigu, pasibaigus sistemą veikiančiam išoriniam trikdančiam poveikiui, sistemą apibūdinantys

parametrai vis labiau nukrypsta nuo nustatytų. Remiantis šiuo apibrėžimu galima teigti, kad autonominės transporto priemonės judėjimas yra nestabilus, jeigu judėjimo metu, nustojus veikti išoriniams veiksniams, esamos realiojo judėjimo koordinatės vis labiau nutolsta nuo pageidaujamų, t. y. didėja koordinatinių nuokrypio vektoriaus elementų reikšmės. Tokiu atveju naudojamas valdymo algoritmas turi įvertinti nuokrypos dydį, t. y. koordinatinių nuokrypio vektoriaus elementų reikšmes ir, atsižvelgiant į taikomą optimizavimo kriterijų ar jų sistemą, priimti racionalų ir patikimą valdymo sprendimą, kuris užtikrintų nuokrypos panaikinimą ar sumažinimą. Taigi yra aišku, kad, siekiant užtikrinti kuo didesnę judėjimo tikslumą ir stabilumą, veikiant įvairiems išoriniams veiksniams, autonominės transporto priemonės analizuojamos kaip tarpdisciplininis objektas, išskiriant ryšius tarp transporto priemonių dinamikos ir automatinio valdymo teorijos.

Chappell (1983) vienas pirmųjų autonominę transporto priemonę aprašė ne kaip vientisą, o kaip kompleksinį tyrimų objektą, sudarytą iš skirtingų sistemų, iš kurių kiekviena atlieka atskiras funkcijas, tačiau sąveikauja tarpusavyje. Tik matematiniu modeliavimu paremtas intelektinis valdymo algoritmas buvo pritaikytas mokslininkų Hogle ir Bonissone (1984). Nors autoriai nurodė, kad, naudojant sudarytą valdymo algoritmą, tam tikrais

atvejais dėl susidūrimo su kelyje buvusiais objektais tirtajai automatizuotai transporto priemonei nepavyko sėkmingai pasiekti numatyto tikslo, vis dėlto intelektiniai valdymo algoritmai palaipsniui tapo plačiai taikomi autonominių transporto priemonių tyrimų srityje. Harris, Read (1988) autonominei transporto priemonei valdyti panaudojo neraiškiosios logikos (angl. *fuzzy logic*), Li, Nutter (1990) – dirbtinių neuroninių tinklų (angl. *neural nets*), Schultz *et al.* (1992) – genetinio programavimo (angl. *genetic algorithm*) metodais pagrįstus algoritmus. Pradėta naudoti skirtingais metodais pagrįstų algoritmų junginius, kurie sudaro hibridines valdymo sistemas (Rastelli *et al.* 2012). Dažnai įvairiuose moksliniuose darbuose (Fang 2004; Bayar *et al.* 2016) tariama, kad autonominės transporto priemonės judėjimas yra tikslus ir stabilus tik tuo atveju, kai judėjimo trajektorijos nuokrypio vektorius yra lygus nuliui. Brown *et al.* (2017) teigia, kad toks griežtas judėjimo nuokrypių apribojimas nėra teisingas, o atitinkamomis sąlygomis valdymo algoritmas turėtų įvertinti, kuriuose trajektorijos taškuose reikia nukrypti nuo pageidaujamos trajektorijos ir kokia nuokrypa yra saugi. Atsižvelgiant į pasirinkamas prielaidas ir taikomus valdymo apribojimus, judėjimo kontroliavimo uždaviniai skirstomi į atitinkamas grupes (Kachroo, Mellodge 2005). Skirtingos kontroliavimo uždavinių grupės gali būti apibendrinamos nusakant tokį patį jų tikslą – užtikrinti tikslų pageidaujamos judėjimo trajektorijos sekimą kontroliuojant skersinį ir / arba išilginį judėjimą (Amer *et al.* 2016). Kontroliavimo uždavinių sprendimas laikomas viena iš sudėtingiausių užduočių, kuriant ir tobulinant autonomines transporto priemones, todėl racionalių valdymo algoritmų formulavimas bei jų analizė yra reikšmingas dėmuo, siekiant užtikrinti sklandesnį, tikslesnį ir saugesnį autonominių transporto priemonių judėjimą įvairiomis trajektorijomis ir sąlygomis.

### Judėjimo trajektorijos optimizavimas

Kaip minėta, racionalaus valdymo algoritmo, kuris užtikrintų tikslų ir stabilų autonominės transporto priemonės judėjimą pageidaujama trajektorija, formulavimas yra daugiausia matematiniu modeliavimu pagrįstas hierarchinis procesas. Šiame darbe teorinis valdymo algoritmo prototipas formuluojamas dviem etapais: pirminis ir antrinis matematinio modeliavimo etapai. Pirminio modeliavimo metu atliekamas teorinio idealaus valdomos fizikinės sistemos matematinio modelio, t. y. pageidaujamos judėjimo trajektorijos ir optimizavimo uždavinio aprašymas (vieno ar kelių kriterijų minimizavimo / maksimizavimo sąlyga ir optimizuojamų parametru parinkimas). Siekiant suformuluoti racionalų valdymo dėsnį, kuris užtikrintų stabilų ir tikslų judėjimą, veikiant išoriniams veiksniams, tariama,

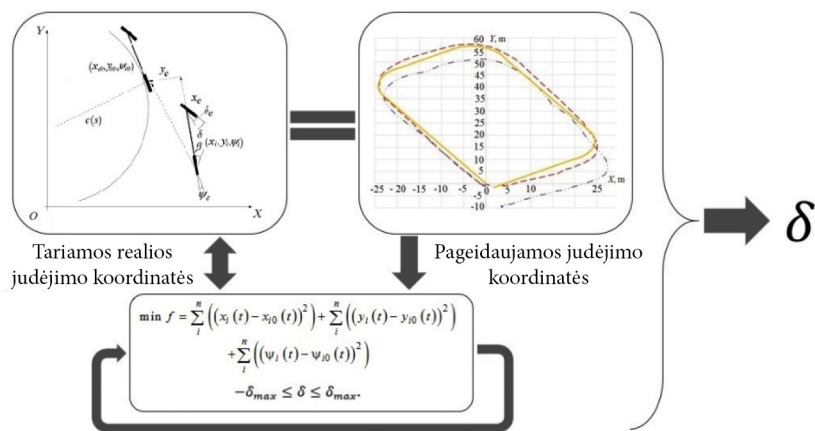
kad optimizuojamas kriterijus yra nuokrypis tarp realių ir pageidaujamų judėjimo koordinatė. Kadangi sprendžiamas optimizavimo uždavinys susijęs su autonominio automobilio valdymu, optimizuojamas parametras – vairuojamųjų ratų pasukimo kampas. Taigi tikslus judėjimo trajektorijos sekimas užtikrinamas taikant skaitinius optimizavimo metodus, kuriais remiantis gali būti valdomas vairuojamųjų ratų pasukimo kampas judėjimo metu.

Antrinis matematinis modeliavimas – teorinė intelektinio valdymo algoritmo prototipo racionalaus veikimo validacija. Tariama, kad antrinio modeliavimo metu sudaromas modelis apibūdina tariamą realų autonominės transporto priemonės judėjimą pageidaujama trajektorija. Pagrindinis pirminio ir antrinio matematinė modelių skirtumas yra tas, kad antrinis matematinis modelis papildomas posistemiu / posistemiais, kurie imituoja išorinių jėgų poveikį judėjimo metu. Tokiu būdu formuluojamo algoritmo veikimas grindžiamas nuolatiniu pirminiu ir antriniu modeliais nustatomų judėjimo koordinatė tarpusavio lyginimu. Jei imituojant išorinių jėgų poveikį, pirminiu ir antriniu modeliais nustatomos judėjimo koordinatės nesutampa, t. y. koordinatė nuokrypio vektorius nėra lygus nuliui, tuomet turi būti sprendžiamas įvardytas optimizavimo uždavinys. Sprendžiant uždavinį nustatoma vairuojamųjų ratų pasukimo kampo reikšmė, kuri taikoma kaip naujas antrinio matematinio modelio įėjimo kintamasis. Optimizavimo procesas gali būti atliekamas neribotą iteracijų skaičių, kol bus nustatyta optimali vairuojamųjų ratų pasukimo kampo reikšmė, kuriai esant koordinatė nuokrypio vektorius tampa lygus nuliui arba sumažėja iki apibrėžtų reikšmių. Aprašyto teorinio algoritmo valdymo dėsnio schema pateikiama 1 pav.

Moksliniuose darbuose autonominių transporto priemonių modeliavimas grindžiamas teiginiu, kad ši yra neholonominė (angl. *non-holonomic*) sistema (Katrakazas *et al.* 2015; Amer *et al.* 2016). Valdymo teorijoje laikoma, kad nagrinėjama sistema yra holonominė tuomet, jeigu jos bendras nepriklausomų kintamųjų – laisvės laipsnių skaičius – atitinka reliatyviųjų judesių, kuriuos galima kontroliuoti, skaičių (Katrakazas *et al.* 2015). Todėl analizuojant neholonominė sistemų judėjimą būtina įvertinti ne tik sistemos koordinatė pokytį, bet ir posūkio kampą apie vertikaliąją ašį. Remiantis šiais teiginiais, atitinkamai suformuluojamas valdymo algoritmo optimizavimo kriterijus:

$$\min f = \sum_i^n \left( (x_i(t) - x_{i0}(t))^2 \right) + \sum_i^n \left( (y_i(t) - y_{i0}(t))^2 \right) + \sum_i^n \left( (\psi_i(t) - \psi_{i0}(t))^2 \right), \quad (1)$$

čia  $x_i, y_i$  – tariosios realiosios judėjimo koordinatės atitinkamai  $x$  ir  $y$  ašyse;  $\psi_i$  – tariamasis pasisukimo kampas apie vertikaliąją ašį;  $x_{i0}, y_{i0}$  – pageidaujamos judėjimo



1 pav. Intelektinio valdymo algoritmo veikimo schema

Fig. 1. The scheme of the operation of the intellectual control algorithm

koordinatės atitinkamai  $x$  ir  $y$  ašyse;  $\psi_{i0}$  – pageidaujamas pasisukimo kampas apie vertikaliąją ašį.

Vienas iš veiksnių, nuo kurio priklauso formuluojamo valdymo algoritmo veikimo racionalumas, yra tinkamas optimizuojamo parametro aprašymas darant apribojimus. Įvertinama, kad optimizuojamas vairuojamųjų ratų pasukimo kampas negali viršyti maksimalaus techniškai galimo vairuojamųjų ratų pasukimo kampo:

$$-\delta_{\max} \leq \delta \leq \delta_{\max}, \quad (2)$$

čia  $\delta$  – vairuojamųjų ratų pasukimo kampas;  $\delta_{\max}$  – maksimalus galimas vairuojamųjų ratų pasukimo kampas.

### Pirminis ir antrinis matematiniai modeliai

Pirminio ir antrinio matematinių modelių kompleksiškus gali apsunkinti valdymo procesą (Kachroo, Mellodge 2005). Todėl šie modeliai sudaromi atsižvelgiant į formuluojamo valdymo algoritmo paskirtį. Formuluojant valdymo algoritmus dažniausiai taikomi kinematinis arba dinaminis transporto priemonės modeliai. Lyginamuosius tyrimus tarp kinematiniais ir dinaminiais modeliais pagrįstų intelektinių valdymo algoritmų atliko Werner *et al.* (2012) ir Taghia *et al.* (2015). Autoriai nurodo, kad iki tam tikro autonominės transporto priemonės judėjimo greičio skirtumas tarp valdymo rezultatų, taikant kinematinį ir dinaminį modelį, nėra ryškus. Werner *et al.* (2012) savo tyrime nustatė, kad kinematinio modeliu pagrįsto valdymo algoritmo naudojimas yra racionalus iki 4,5 m/s judėjimo greičio ribos, Taghia *et al.* (2015) nurodyta riba – 3 m/s. Šiame darbe analizuojamas autonominio automobilio judėjimas, kai jo greitis yra  $1 \text{ m/s} \leq v \leq 3 \text{ m/s}$ , todėl, remiantis įvardytų autorių darbais, pirminio ir antrinio modeliavimo metu sudaromi kinematiniai transporto priemonės modeliai.

Sudarant kinematinis modelius, įvertinamos autonominės transporto priemonės judėjimo koordinatės, judėjimo greitis ir pagreitis bei šių parametų tarpusavio ryšiai. Siekiant sumažinti valdymui reikalingų kompiuterinių išteklių poreikį ir palengvinti valdymo uždavinio sprendimą, moksliniuose darbuose keturratis autonominės transporto priemonės kinematinis modelis dažnai supaprastinamas į dviratį (Bayar *et al.* 2016; Brown *et al.* 2017). Atitinkamai ir šiame darbe naudojamas supaprastintas – dviratis – transporto priemonės kinematinis modelis (2 pav.). Bendroju atveju tiek autonominės transporto priemonės tariamo realaus judėjimo vektorius, tiek pageidaujamos judėjimo trajektorijos koordinačių vektorius yra išreiškiami (Wang *et al.* 2014):

$$\{q_i\}^T = [x_i \quad y_i \quad \psi_i]. \quad (3)$$

Autoriai LaValle (2006) ir Siegwart *et al.* (2011) savo darbuose nurodo, kad kinematiškai modeliuojant neholonomines sistemas, negalima įvertinti judėjimo slydimo režimu, todėl erdvė, kurioje vyksta tyrimas, apibrėžiama kaip ideali. Tariama, kad autonominės transporto priemonės ratas visada rieda neslysdamas absoliučiai standžiu šiurkščiu paviršiumi. Pagrindinė šios prielaidos priežastis – neholonominių sistemų suvaržymas diferencialiniais ryšiais. Svarbu tai, kad padanga yra elastingas besideformuojantis kūnas, o jos ir kelio sąlyčio zonoje veikia sankybio jėgos, todėl grynojo rato riedėjimo be slydimo prielaida modeliuojant nėra racionali. Moksliniuose darbuose iš dalies ši problema sprendžiama kinematiname modelyje įvertinant įstrižojo riedėjimo kampą transporto priemonės sunkio centre ar kitame pasirinktame taške (Rajamani 2012; Wang *et al.* 2014). Taikant kinematinį modelį, realaus judėjimo koordinačių kitimas yra charakterizuojamas taip (Rajamani 2012):

$$\dot{x} = v \cdot \cos(\psi + \beta), \quad (4)$$

$$\dot{y} = v \cdot \sin(\psi + \beta), \quad (5)$$

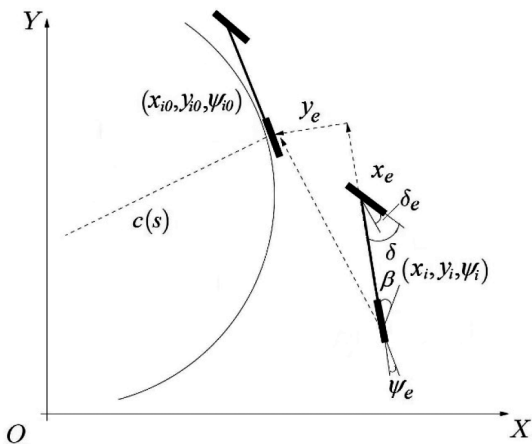
$$\dot{\psi} = \frac{v \cdot \cos(\beta)}{L} \cdot \operatorname{tg}(\delta_p), \quad (6)$$

čia  $v$  – judėjimo greitis;  $\beta$  – įstrižojo riedėjimo kampas sunkio centre;  $L$  – ratų bazė;  $\delta_p$  – priekinių vairuojamųjų ratų pasukimo kampas.

Įstrižojo riedėjimo kampas sunkio centre išreiškiamas (Rajamani 2012):

$$\beta = \operatorname{tg}^{-1} \left( \frac{l_g \cdot \operatorname{tg}(\delta_p)}{L} \right), \quad (7)$$

čia  $l_g$  – atstumas nuo galinės automobilio ašies iki sunkio centro.



2 pav. Autonominės transporto priemonės dviračio kinematinio modelio schema (sudaryta pagal Fang 2004)

Fig. 2. The scheme of the kinematic bicycle model of the autonomous vehicle (based on Fang 2004)

Fang (2004) įvertino kinematinio modelio geometrinius ryšius ir tiksliai suformulavo koordinatinių nuokrypio vektorių:

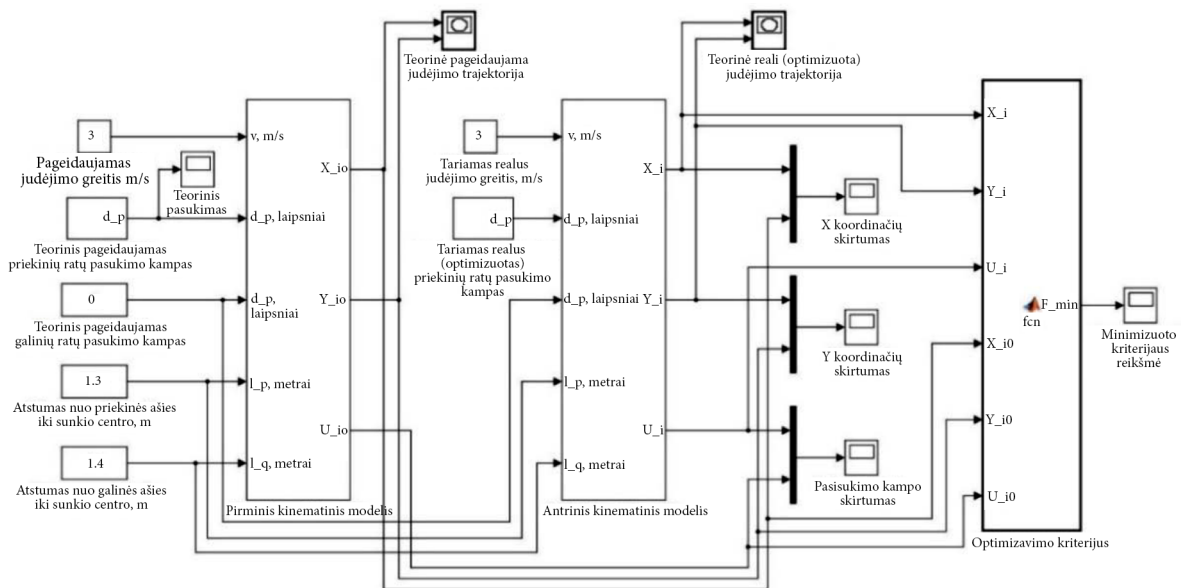
$$\{q_e\} = \begin{bmatrix} x_e \\ y_e \\ \psi_e \end{bmatrix} = \begin{bmatrix} \cos(\psi) & \sin(\psi) & 0 \\ -\sin(\psi) & \cos(\psi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x_{i0} - x_i \\ y_{i0} - y_i \\ \psi_{i0} - \psi_i \end{bmatrix}. \quad (9)$$

čia  $x_e$  – koordinatinių nuokrypis  $x$  ašyje;  $y_e$  – koordinatinių nuokrypis  $y$  ašyje;  $\psi_e$  – pasisukimo kampo apie vertikaliąją ašį nuokrypis.

### Teorinė formuluojamo valdymo algoritmo validacija

Siekiant atlikti teorinę validaciją, aprašytas valdymo algoritmo prototipas, t. y. pirminis bei antrinis kinematiniai modeliai ir optimizavimo uždavinys, buvo sudaryti programinio paketo *MATLAB / Simulink* aplinkoje (3 pav.).

Vienas iš teorinės validacijos tikslų – įvertinti metodų tinkamumą, kuriais remiantis būtų galima suderinti anksčiau teoriškai aprašyto optimizavimo uždavinio ir kinematinų modelių sprendimo algoritmus. Dėl šios priežasties teorinė validacija atliekama dviem etapais. Pirmojo validacijos etapo metu sąsaja tarp įvardytų sprendimo algoritmų suformuluojama, t. y. uždavinių sprendimo algoritmai suderinami, naudojant programinio paketo *MATLAB / Simulink* parametrų optimizavimo įrankį. Optimizavimo uždavinys sprendžiamas taikant simplekso metodą. Antrojo validacijos etapo metu parametrų optimizavimo įrankis nenaudojamas, optimizavimo uždaviniui spręsti taikomas ir formuluojamas genetinio programavimo algoritmas. Taikant genetinio



3 pav. Sudarytas valdymo algoritmas programinio paketo *MATLAB / Simulink* aplinkoje

Fig. 3. Designed control algorithm in the environment of the software package *MATLAB / Simulink*



```

%% Function
function f = OptimizavimoKriterijus(d)
f = ((x_i - x_i0)^2) + ((y_i - y_i0)^2) + (((v*cos(atan(l_g*tan(d)/l))/L) *
tan(d))^2 * t) - u_i0^2; %% Aprasomas optimizavimo kriterijus
end

%% Script
FitFcn = @OptimizavimoKriterijus;
navrs = 1;
[d, fval] = qa(FitFcn,navrs); %% Taikomas genetinio programavimo metodas

```

4 pav. Suformuluoto genetinio programavimo algoritmo fragmentas programinio paketo *MATLAB* aplinkoje  
 Fig. 4. A part of the applied method of the genetic algorithm in the environment of the software package *MATLAB*

programavimo algoritmą, išraiška (6) yra integruojama (integravimo kintamasis – laikas  $t$ ), o gautoji reikšmė naudojama pertvarkant optimizavimo kriterijų (1). Suformuluotas genetinio programavimo algoritmas ir pertvarkytas optimizavimo kriterijus pateikiami 4 pav.

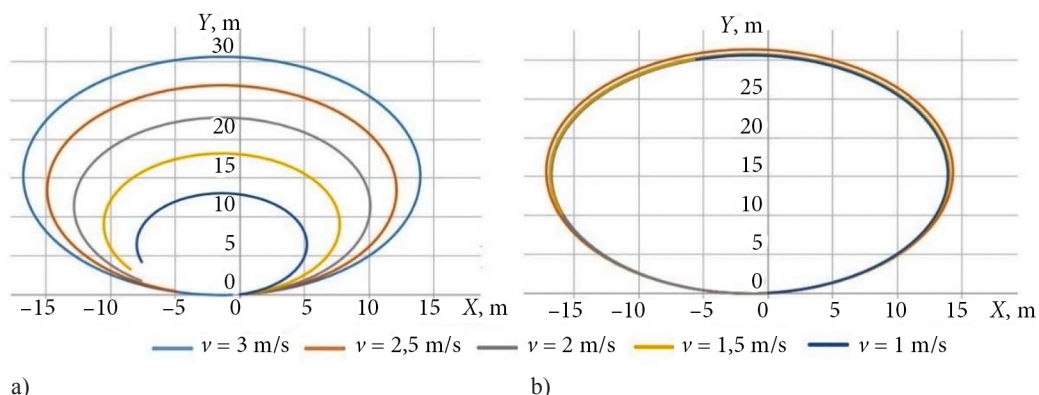
Kadangi šiame straipsnyje formuluojamas teorinis valdymo algoritmo prototipas, tariama, kad validacija atliekama autonominei transporto priemonei atliekant techniškai nesudėtingą manevrą: tiek pirmojo, tiek antrojo validacijos etapų metu autonominis automobilis juda ratu esant pastoviam vairuojamųjų ratų pasukimo kampui ir iki galo apvažiuo viena ratą sustoja (5 pav.). Pagrindiniai pirminio ir antrinio kinematinė modelių įvesties duomenys:  $L = 2,7$  m,  $l_g = 1,4$  m,  $\beta_{max} = 27^\circ$ , teorinis pageidaujamas judėjimo greitis  $v = 3$  m/s, vairuojamųjų ratų pasukimo kampas  $\delta_p = 10^\circ$ . Valdymo algoritmo racionalaus veikimo validacija atliekama laikantis sąlygos, kad tariamas realus autonominio automobilio greitis dėl veikiančių išorinių veiksnių yra mažesnis už teorinį pageidaujamą judėjimo greitį: 2,5 m/s, 2 m/s, 1,5 m/s, 1 m/s.

### Rezultatų analizė

Pirmojo ir antrojo validacijos etapų rezultatai pateikiami 5 pav., atitinkamai a ir b dalyse. Išanalizavus 5 pav. b dalyje pateiktus pirmojo validacijos etapo rezultatus, matoma, kad tikslas – užtikrinti sklandų pageidaujamos

judėjimo trajektorijos sekimą optimizuojant vairuojamųjų ratų pasukimo kampą – nepasiektas. Valdymo algoritmo veikimas taip pat nėra racionalus, nes kuo tariamas realusis judėjimo greitis mažesnis, tuo skirtumas tarp pageidaujamos ir tariamos realiosios judėjimo trajektorijų didėja. Šis reiškinys gali būti paaiškintas tuo, kad, naudojant programinio paketo *MATLAB / Simulink* parametų optimizavimo įrankį, vairuojamųjų ratų pasukimo kampo reikšmė optimizuojama, siekiant surasti judėjimo trajektorijos formą, artimiausią pageidaujamai, kurią automobilis gali įveikti esant tam pačiam judėjimo (simuliacijos) laikui, kai judėjimo greitis yra mažesnis. Tai reiškia, kad aprašytam optimizavimo uždaviniui suderinti su kinematinė modelių sprendimo algoritmu taikant programinio paketo *MATLAB / Simulink* parametų optimizavimo įrankį, pageidaujama judėjimo trajektorija nėra sekama, ir toks valdymo algoritmas negali būti taikomas saugiam, tiksliam ir stabiliam autonominės transporto priemonės judėjimui užtikrinti. Kitus tokio valdymo algoritmo išvesties parametrus analizuoti nėra tikslinga.

Priešingai nei pirmajame etape, antrojo validacijos etapo metu judėjimo trajektorijos kitimas nėra pastebimas. Šiuo atveju, siekiant įvertinti formuluojamo valdymo algoritmo prototipo veikimo racionalumą, kritiškai svarbu tai, kad taikant įvardytą valdymo algoritmą atliekamas judėjimo trajektorijos sekimas yra palyginti tikslus (5 pav., b dalis).



5 pav. Teorinės validacijos rezultatai: a) pirmojo validacijos etapo rezultatai; b) antrojo validacijos etapo rezultatai

Fig. 5. Results of the theoretical validation: a) results of the first stage of the validation; b) results of the second stage of the validation

Optimizuotas vairuojamųjų ratų pasukimo kampas esant 2,5 m/s judėjimo greičiui – 9,77°, 2 m/s – 9,94°, 1,5 m/s – 9,97°, 1 m/s – 10,03°. Taip pat, mažėjant judėjimo greičiui, atitinkamai buvo nuvažiuojamas ir trumpesnis atstumas. Remiantis šiais rezultatais galima teigti, kad visais antrojo validacijos etapo metu analizuotais judėjimo atvejais tikslas užtikrinti sklandų pageidaujamos judėjimo trajektorijos sekimą optimizuojant vairuojamųjų ratų pasukimo kampą pasiektas, o suformuluotas genetinio programavimo algoritmas (4 pav.) užtikrina racionalų teorinio valdymo algoritmo prototipo veikimą. Vis dėlto šiame valdymo algoritmo prototipo kūrimo etape, nuosekliai veikianti sąsaja tarp genetinio programavimo ir kinematinų modelių sprendimo algoritmų nebuvo sudaryta. Tai reiškia, kad, norint gauti validavimo rezultatus, genetinio programavimo algoritmas turėjo būti pritaikomas iš naujo kiekvienu laiko momentu. Formuluoiant teorinį valdymo algoritmo prototipą toks validavimo duomenų gavimo ir apdorojimo būdas yra pakankamas, tačiau, siekiant toliau analizuoti ir tobulinti formuluojamo algoritmo veikimą, būtina sudaryti nuosekliai ir efektyviai veikiančią sąsają tarp genetinio programavimo ir kinematinų modelių sprendimo algoritmų.

## Išvados

Straipsnyje sudarytas ir aprašytas teorinis valdymo algoritmo prototipas, kuriuo, optimizuojant transporto priemonės vairuojamųjų ratų pasukimo kampą, turi būti užtikrinamas pageidaujamos judėjimo trajektorijos sekimas. Atlikus sudaryto teorinio valdymo algoritmo prototipo racionalaus veikimo analizę nustatyta, kad:

1. Optimizavimo uždavinio ir kinematinų modelių sprendimo algoritmams suderinti, naudojant programinio paketo *MATLAB / Simulink* parametrų optimizavimo įrankį ir simplekso metodą, tikslas užtikrinti sklandų pageidaujamos judėjimo trajektorijos sekimą nepasiektas, o valdymo algoritmo veikimas nėra racionalus. Kuo tariamas realusis judėjimo greitis mažesnis, tuo skirtumas tarp pageidaujamos ir tariamos realios judėjimo trajektorijų didėja. Toks valdymo algoritmas negali būti taikomas norint užtikrinti saugų, tikslų ir stabilų autonominės transporto priemonės judėjimą.
2. Optimizavimo uždaviniui spręsti suformulavus ir pritaikius genetinio programavimo algoritmą, atliekamas judėjimo trajektorijos sekimas yra palyginti tikslus. Paklaida tarp pageidaujamo ir optimizuoto vairuojamųjų ratų pasukimo kampo neviršija 2,3 %. Tikslas užtikrinti sklandų pageidaujamos judėjimo trajektorijos sekimą pasiektas.

3. Nuosekliai ir efektyviai veikianti sąsaja tarp genetinio programavimo ir kinematinų modelių sprendimo algoritmų nebuvo sudaryta. Norint gauti validavimo rezultatus, genetinio programavimo algoritmas turėjo būti pritaikomas iš naujo kiekvienu laiko momentu, todėl, toliau tobulinant formuluojamo valdymo algoritmo veikimą, pirmiausia būtina sudaryti nuosekliai ir efektyviai veikiančią sąsają tarp genetinio programavimo ir kinematinų modelių sprendimo algoritmų.

## Literatūra

- Amer, N. H.; Zamzuri, H.; Hudna, K.; Kadir, Z. A. 2016. Modelling and control strategies in path tracking control for autonomous ground vehicles: a review of state of the art and challenges, *Journal of Intelligent & Robotic Systems* 86(2): 225–254. <https://doi.org/10.1007/s10846-016-0442-0>
- Anderson, J. M.; Kalra, N.; Stanley, K. D.; Sorensen, P.; Samaras, C.; Oluwatola, O. A. 2016. *Autonomous vehicle technology. A guide for policymakers*. RAND Corporation. <https://doi.org/10.7249/RR443-2>
- Bayar, G.; Bergerman, M.; Konukseven, E. I.; Koku, A. B. 2016. Improving the trajectory tracking performance of autonomous orchard vehicles using wheel slip compensation, *Biosystems Engineering* 146: 149–164. <https://doi.org/10.1016/j.biosystemseng.2015.12.019>
- Brown, M.; Funke, J.; Erlien, S.; Gerdes, J. C. 2017. Safe driving envelopes for path tracking in autonomous vehicles, *Control Engineering Practice* 61: 307–316. <https://doi.org/10.1016/j.conengprac.2016.04.013>
- Chappell, S. 1983. Software organization in an autonomous vehicle, in *Proceedings of the 3rd International Symposium on Unmanned Untethered Submersible Technology* 3: 162–169. <https://doi.org/10.1109/UUST.1983.1158513>
- Fang, H. 2004. *Automatic guidance of farm vehicles in presence of sliding effects: A Scientific Report for the Post-Doc Research Project of All-Terrain Autonomous Vehicle Control*.
- Harris, C. J.; Read, A. B. 1988. Knowledge based fuzzy motion control of autonomous vehicles, in *Proceedings of the IFAC Workshop* 21(13): 139–144.
- Hogle, A. R.; Bonissone, P. P. 1984. A fuzzy algorithm for path selection in autonomous vehicle navigation, in *The 23rd IEEE Conference on Decision and Control*: 898–900.
- Januševičius, V. S. 2003. *Automatinis valdymas: teorija, uždaviniai, sprendimai*. Kaunas: Technologija.
- Kachroo, P.; Mellodge, P. 2005. *Mobile robotic car design*. McGraw Hill Professional.
- Kalra, N.; Paddock, S. M. 2016. Driving to safety. *How many miles of driving would it take to demonstrate autonomous vehicle reliability*. Rand Corporation. <https://doi.org/10.7249/RR1478>
- Katrakazas, C.; Quddus, M.; Chen, W. H.; Deka, L. 2015. Real-time motion planning methods for autonomous on-road driving: state-of-the-art and future research directions, *Transportation Research Part C* 60: 416–442. <https://doi.org/10.1016/j.trc.2015.09.011>

- LaValle, S. M. 2006. *Planning algorithms*. Cambridge University Press. <https://doi.org/10.1017/CBO9780511546877>
- Li, P.; Nutter, R. S. 1990. A landmark recognition system using artificial neural nets, in *IEEE International Conference on Systems Engineering*, 14 June 2012, Pittsburgh, PA, USA, 573–576. <https://doi.org/10.1109/ICSYSE.1990.203222>
- Rajamani, R. 2012. *Vehicle dynamics and control*. Springer. <https://doi.org/10.1007/978-1-4614-1433-9>
- Rastelli, J. P.; Gajate, A.; Milanés, V.; Onieva, E.; Santos, M. 2012. Design and implementation of a neuro-fuzzy system for longitudinal control of autonomous vehicles, in *IEEE International Conference on Fuzzy Systems*, 18–23 July, Barcelona, Spain. 7 p.
- Schultz, A. C.; Grefenstette, J. J.; De Jong, K. A. 1992. Adaptive testing of controllers for autonomous vehicles, in *Proceedings of the 1992 Symposium on Autonomous Underwater Vehicle Technology*, 06 August 2002, Washington, DC, USA, USA, 158–164. <https://doi.org/10.1109/AUV.1992.225178>
- Siegwart, R.; Nourbakhsh, I. R.; Scaramuzza, D. 2011. *Introduction to autonomous mobile robots*. MIT Press.
- Taghia, J.; Wang, X.; Lam, S.; Katupittya, J. 2015. A sliding mode controller with a nonlinear disturbance observer for a farm vehicle operating in the presence of wheel slip, *Autonomous Robots* 41(1): 71–88. <https://doi.org/10.1007/s10514-015-9530-4>
- Wang, X.; Fu, M.; Ma, H.; Yang, Y. 2014. Lateral control of autonomous vehicles based on fuzzy logic, *Control Engineering Practice* 34: 1–17. <https://doi.org/10.1016/j.conengprac.2014.09.015>
- Werner, R.; Mueller, S.; Kormann, G. 2012. *Path tracking control of tractors and steerable towed implements based on kinematic and dynamic modeling*. International Society of Precision Agriculture. 16 p.

## FORMULATION AND THEORETICAL ANALYSIS OF THE CONTROL ALGORITHM OF THE AUTONOMOUS VEHICLE

P. Skačkauskas, V. Mejeris, E. Sokolovskij

Abstract

With the technologies of the autonomous vehicles improving, one of the biggest tasks is safe and effective integration of the autonomous vehicles into the transport system. The safety of such vehicles depends on their ability to move stably and accurately in the desired trajectory. The intellectual control algorithm is the most important system, controlling the movement of the autonomous vehicle. That is why the formulation and analysis of the rational intellectual control algorithms based on different methods is a significant element when seeking that the autonomous vehicles become a valuable road user. A theoretical version of an intellectual control algorithm, which is designed to ensure an accurate following of the desired movement trajectory under different movement conditions, is presented and described in the work. The operation of the algorithm is based on the comparison of the mathematical models of two vehicles and the solution of the optimization task. The described control algorithm is presented in the environment of the software package *MATLAB/Simulink*, the analysis of the rational operation of the designed algorithm is done.

**Keywords:** autonomous vehicle, genetic algorithm, movement trajectory following, deviation.