

**KLAIPĖDOS UNIVERSITETAS**

Jūrų technikos fakultetas

Informatikos inžinerijos katedra

**MANIPULATORIAUS IŠMANIOSIOS VALDYMO  
SISTEMOS KŪRIMAS TAIKANT DAUGIAKRITERINIUS  
SPRENDIMO PRIĖMIMO METODUS**

**DEVELOPMENT OF MANIPULATOR SMART CONTROL  
SYSTEM APPLYING MULTIPLE CRITERIA DECISION  
SUPPORT METHODS**

Techninių informacinių sistemų inžinerijos specialybės magistro baigiamasis darbas

Autorius: TMII-11 Tomas Šarpnickas

Vadovas: doc. dr. Violeta Bulbenkienė

Klaipėda, 2013

## ANOTACIJA

Išmanūs ir manevringi manipulatoriai, gebantys patys tinkamai koordinuoti savo veiksmus, yra plačiai taikomi didžiuosiuose pasaulio uostuose, geležinkeliuose bei įvairiose gamybos ir mokslo srityse. Nagrinėjant manipulatoriaus kinematikos uždavinius, būtina apskaičiuoti lanksčiai sujungtų svirčių trajektorijas, atsižvelgiant į objekto koordinatų kitimo dėsningumus. Manipulatoriaus modelio kinematikos lygčių užrašymas ir jų analizė suteikia galimybę sudaryti programos algoritmą svirčių kampų skaičiavimui ir judančiųjų dalių trajektorijos optimizavimui, siekiant sumažinti kelio ilgį ir efektyviau panaudoti manipulatoriaus variklius. Manipulatoriaus svirčių judėjimo trajektorijoje atsiradus kliūčiai, ji yra analizuojama ir reikalui esant pateikiama nauja svirčių trajektorija kliūčiai apeiti. Šiame darbe analizuojama manipulatoriaus kinematika bei pateikiamas algoritmas skirtas judančių dalių trajektorijų optimizavimui, siekiant sumažinti energetinius kaštus bei išanalizuoti manipulatoriaus judėjimo trajektorijoje iškilusią kliūtį ir jos išvengti.

**Raktiniai žodžiai:** manipulatorius, išmaniosios sistemos, algoritmas, programavimas.

## ABSTRACT

Smart and manoeuvrable manipulators, which are able to decide their movements, are widely applied in the world's biggest ports, railway stations and in manufacturing and science. Analysis of manipulator kinematics needs for compiling of arm movement equations which allow specifying requirements with the view of the optimized trajectory and more effective usage of motors. When an obstacle occurs in the manipulators levers trajectories, obstacles position is analyzed and if necessary new levers trajectories is created to avoid obstacles. The thoughtful research enables to create an algorithm for computation of angles of the arms by taking into account the conditions determined by model of manipulator. In this paper inverse kinematics of manipulator is analysed and algorithm is created for trajectory optimization to reduce energy costs and for analyzing obstacles position in the manipulators movement trajectory to circumventing obstacle.

**Keywords:** manipulator, smart systems, algorithm, programing.

## SUMMARY

Smart and manoeuvrable manipulators, which are able to decide their movements, are widely applied in the world's biggest ports, railway stations and in manufacturing and science. The main problem of creating smart manipulator systems is analysis of manipulator kinematics, which is needed for compiling of arm movement equations which allow specifying requirements with the view of the optimized trajectory and more effective usage of motors. Many accidents happen of operator controlled manipulator by chashing it to obstacles. In this work three levers redundant manipulator is analyzed which allows to avoid existing obstacles. When an obstacle occurs in the manipulators levers trajectories, in this thesis obstacles position is analyzed and if necessary new levers trajectories is created to avoid obstacles. Obstacle position is analyzed to determine if obstacle prevent manipulators grabber to reach task. If obstacle interfere grabber to reach task - new intermediate points in grabbers trajectory are created for avoiding obstacles. The thoughtful research enables to create an algorithm for computation of angles of the arms by taking into account the conditions determined by model of manipulator. In this paper inverse kinematics of manipulator is analysed and algorithm is created for trajectory optimization to reduce energy costs and for analyzing obstacles position in the manipulators movement trajectory to circumventing obstacle. To prove manipulators smart control system is effective and safe by avoiding obstacles "Matlab" simulation was presented.

Šarpnickas T. Development of manipulator smart control system applying multiple criteria decision support methods. Techninių informacinių sistemų inžinerijos magistro baigiamasis darbas. Darbo vadovas doc. dr. V. Bulbenkinė, Klaipėdos universitetas: Klaipėda, 2013., p-53.

## SANTRUMPŲ IR TERMINŲ ŽODYNĖLIS

### A

**AI** (angl. *artificial intelligent*) – dirbtinis intelektas.

**Atvirkštinė kinematika** (angl. *inverse kinematics*) – Manipulatoriaus svirčių tarpusavės padėties kampų skaičiavimas pagal griebtuvo koordinates.

**Autonominis** (angl. *autonomous*) – robotas ar įrenginys savarankiškai, be visiško žmogaus įsikišimo, atliekantis veiksmus.

### C

**C++** - bendros paskirties programavimo kalba.

### D

**DB** (angl. *data base*) – duomenų bazė.

**DOF** (angl. *degrees of freedom*) – laisvės laipsniai.

### I

**IS** (angl. *information system*) – informacinė sistema

**IF/THEN** – Programavimo taisyklės – Jeigu „veiksmas“, tai „atoveiksmis“.

**Išmanioji sistema** (angl. *smart system*) – apibrėžiama kaip nedidelis įtaisas, kuris valdo jutimo, paleidimo ir kontrolės funkcijas. Išanalizavus aplinkos situacija priima atitinkamus veiksmus.

### J

**Jutiklis** (angl. *sensor*) – įtaisas skirtas registruoti parametrus pagal savo prigimtį.

### M

**Manipulatorius** (angl. *manipulator*) – robotinė ranka, imituojant žmogaus rankos veiksmus, atliekanti tam tikras užduotis.

**Matlab** - yra daugiaplatformė MathWorks programinė įranga, skirta įvairių mokslo šakų problemoms spręsti, ypač matematinėms.

### O

**Operatorius** (angl. *operator*) – specialistas, atliekantis kokią nors operaciją.

### P

**PLV** (angl. *Programmable logic controller*) – programuojamas loginis valdiklis

**Pagalbinės sistemos** (angl. *assistance systems*) – integruota sistema, pritaikyta padėti efektyviau žmogui atlikti tam tikras funkcijas.

## R

**Realaus laiko sistemos** (angl. *online systems*) – tai sistemos galinčios atlikti operacijas tuo laiko momentu. (realiu laiku).

## T

**Tiesioginė kinematika** (angl. *forward kinematics*) – Manipulatoriaus griebtuvo koordinačių radimas pagal svirčių tarpusavės padėties kampus.

**TIS** – Techninė informacinė sistema.

## V

**Valdiklis** (angl. *controller*) - specializuotos paskirties nesavarankiškas kompiuteris, valdymo įtaisas.

## Ž

**Žingsninis variklis** (angl. *step motor*) – tai elektros mašina, pakeičianti valdymo impulsus į kampinį arba linijinį poslinkį.

## TURINYS

SUMMARY .....	4
ĮVADAS .....	12
I. ANALITINĖ (LITERATŪROS ANALIZĖS) DALIS .....	13
1. Manipulatoriaus valdymas.....	13
2. Manipulatoriaus svirčių kinematinė analizė .....	14
2.1. Atvirkštinė kinematika .....	15
3. Manipulatoriaus griebtuvo trajektorija .....	16
3.1. Manipulatoriaus griebtuvo trajektorijos planavimas.....	16
3.2. Manipulatoriaus griebtuvo trajektorijos optimizavimas .....	17
4. Kliūčių išvengimas .....	18
II. METODINĖ DALIS .....	21
1. Manipulatoriaus kūrimo problematika .....	21
2. Manipulatoriaus kūrimo sprendimas .....	22
3. Žingsninių variklių analizė .....	23
3.1. Žingsniniai varikliai ir jų parametrai.....	23
3.2. Valdymo pobūdis .....	24
III. PROJEKTINĖ DALIS.....	26
1. Atvirkštinės kinematikos sprendimas.....	26
1.1. Manipulatoriaus veikimo principas.....	26
1.2. Svirčių kampinės padėties skaičiavimas .....	27
1.3. Kampai aprašantys manipulatoriaus svirčių judesių koordinavimą.....	28
2. Tiesinės kinematikos sprendimas .....	29
3. Kliūčių pozicijos analizė ir išvengimo metodas .....	31
3.1. Kliūčių pozicijos įvertinimas .....	32
3.2. Kliūčių išvengimo metodas.....	36
4. Manipulatoriaus valdymo sistemos algoritmas .....	40
4.1. Atvirkštinės kinematikos skaičiavimų algoritmas.....	40

4.2.	Tiesinės kinematikos skaičiavimų algoritmas.....	42
4.3.	Kliūčių pozicijos įvertinimo algoritmas.....	43
4.4.	Kliūčių išvengimo algoritmas .....	44
4.5.	Manipulatoriaus valdymo sistemos algoritmas.....	45
IV.	EKSPERIMENTINĖ DALIS .....	47
1.	Manipulatoriaus eksperimentiniai tyrimai.....	47
	IŠVADOS.....	49
	LITERATŪRA .....	50
	PRIEDAI .....	53
1.	Mokslinės publikacijos.....	54
2.	Kompaktinė plokštelė.....	64

## LENTELIŲ SĄRAŠAS

Lentelė 1. Pilno žingsnio impulsų seka .....	25
Lentelė 2. Dviejų polių variklio impulsai .....	25
Lentelė 3. Manipulatoriaus svirčių judesių koordinavimą aprašantys kampai.....	29
Lentelė 4. $AKA_1K_1$ kampų koordinatės .....	32
Lentelė 5. Kliūties padėties variantai svirties AK atžvilgiu .....	36
Lentelė 6. Griebtuvo tarpinių taškų koordinatės .....	40

## ILIUSTRACIJŲ SĄRAŠAS

1 pav. Medienos rąstų krovimo manipulatoriaus modelis.....	13
2 pav. Tiesinės ir atvirkštinės kinematikos ryšiai.....	14
3 pav. Svirčių padėtys taikant gradiento projekcijos metodą [5] .....	15
4 pav. Manipulatoriaus užduoties sprendimo būdai [6] .....	16
5 pav. COMAU SMART3-S2 manipulatorius [29] .....	16
6 pav. Griebtuvo judėjimo trajektorijos [9] .....	17
7 pav. Teleskopinis manipulatorius [19] .....	18
8 pav. Trijų svirčių manipulatoriaus eskizas [25] .....	18
9 pav. Kliūčių išvengimo trajektorija [25].....	19
10 pav. Kliūtis atstumas iki manipulatoriaus [26].....	19
12 pav. Žingsninis variklis turintis tris išvadus .....	23
13 pav. Žingsninio variklio apvijos atskiruose korpusuose.....	24
14 pav. Dviejų polių žingsninis variklis .....	24
15 pav. Daugiapolis žingsninis variklis.....	24
16 pav. Manipulatoriaus modelis .....	26
17 pav. Svirčių tarpusavės padėties variantai .....	26
18 pav. Manipulatoriaus ribiniai kampai.....	27
19 pav. Svirčių kampinės padėtys .....	27
20 pav. Tiesinės kinematikos skaičiavimai .....	30
21 pav. Kliūčių pozicijos manipulatoriaus judėjimo trajektorijoje .....	31
22 pav. Manipulatoriaus svirties AK judėjimo trajektorija .....	32
23 pav. Kliūtis pozicijos tikrinimas funkcijų $f_1(x)$ ir $f_2(x)$ atžvilgiu.....	34
24 pav. Kliūtis pozicijos tikrinimas funkcijų $f_3(x)$ ir $f_4(x)$ atžvilgiu.....	35
25 pav. Kliūtis manipulatoriaus svirties trajektorijoje .....	36
26 pav. Kliūtis išvengimo metodas.....	37
27 pav. Griebtuvo trajektorijos tarpiniai taškai .....	37
28 pav. Tarpinio taško $K_2$ koordinačių radimas .....	38
29 pav. Tarpinio taško $K_3$ koordinačių radimas .....	39
30 pav. Atvirkštinės kinematikos skaičiavimų algoritmas .....	41
31 pav. Tiesinės kinematikos skaičiavimų algoritmas .....	42
32 pav. Kliūtis pozicijos įvertinimo algoritmas.....	43
33 pav. Kliūtis išvengimo algoritmas .....	45

34 pav. Manipulatoriaus valdymo sistemos algoritmas .....	46
35 pav. Manipulatoriaus užduoties atlikimas „Matlab“ .....	47
36 pav. Manipulatoriaus užduoties atlikimas esant kliūčiai „Matlab“ .....	48

## ĮVADAS

Manipuliatorių galima apibrėžti kaip įrenginį, kuris gali atlikti sudėtingą erdvinį judesį, imituojantį žmogaus rankos veiksmus [1, 2]. Įvairių konstrukcijų manevringi, intelektualūs ir tikslūs manipulatoriai, gebantys patys tinkamai koordinuoti savo veiksmus, yra plačiai taikomi uostuose, geležinkeliuose bei įvairiose gamybos ir mokslo srityse [3]. Pažangūs autonominiai manipulatoriai vis labiau integruojami įvairiose srityse, kur reikalinga greitateika, tikslumas bei didelė fizinė jėga. Siekiant išvengti žmogiškojo faktoriaus klaidų ir padidinti sistemos veikimo tikslumą, manipulatoriams yra kuriamos savarankiškos valdymo sistemos, kurios priima sprendimus ir koordinuoja manipuliatorių.

Siekiant užtikrinti efektyvų techninių sistemų valdymą, ypač svarbu tinkamai aprašyti sistemos veikimo algoritmą, leidžiantį sukurti išmanios autonominės sistemos valdomą manipuliatorių, kuris būtų taikomas įvairiose srityse [4]. Viena iš svarbiausių manipulatoriaus kūrimo problemų yra siejama su aktualia atvirkštinės kinematikos užduotimi - lanksčiai sujungtų svirčių kampų skaičiavimu, norint tinkamai valdyti tų svirčių padėtį erdvėje pagal nustatytas darbo įrankio koordinatas ir jo pozicionavimo sąlygas [5, 6]. Kita ne mažiau svarbi problema yra susijusi su svirčių tarpusavės padėties kampų konfigūracijos parinkimu, nuo kurio priklauso energetiškai efektyvus manipulatoriaus variklių darbas [8, 9]. Autonominėms informacinėms – techninėms sistemos taikomi ypatingai aukšti reikalavimai saugumui [10]. Atsiradus manipulatoriaus veikimo zonoje nenumatytų kliūčių, valdymo sistema turi atitinkamai reaguoti ir jų išvengti [11]. Šiame darbe yra analizuojama atsiradusių kliūčių pozicija manipulatoriaus atžvilgiu bei pateikiamas kliūčių išvengimo metodas. Integravus šį metodą į manipulatoriaus valdymo sistemą - išpildomi visi reikalavimai manipulatoriaus išmaniajai valdymo sistemai kurti.

Nagrinėjant šių uždavinių sprendimo galimybes, buvo suprojektuotas virtualus manipulatoriaus eksperimentinis modelis, skirtas svirčių kinematikos matematinei analizei atlikti ir optimaliam valdymo algoritmui sudaryti bei išvengti kliūčių, atsiradusių manipulatoriaus svirčių judėjimo trajektorijoje.

**Šio darbo tikslas** – sukurti energetiškai efektyvią išmaniają manipulatoriaus valdymo sistemą bei padidinti sistemos saugumą taikant kliūčių analizės ir išvengimo algoritmus.

Siekiant pasiekti užsibrėžtą tikslą, suformuluoti šie **darbo uždaviniai**:

- Apžvelgti esamus manipulatoriaus valdymo sistemų analogus ir įvertinti jų galimybes.
- Remiantis atlikta analize, sudaryti alternatyvius skaičiavimus bei optimalius algoritmus, siekiant išspręsti kilusias išmaniosios valdymo sistemos kūrimo problemas;
- Sukurti ir pritaikyti manipulatoriaus sistemai kliūčių išvengimo metodą.
- Eksperimentiškai pagrįsti sukurtos sistemos funkcionalumą, atlikti sistemos verifikacija.

## I. ANALITINĖ (LITERATŪROS ANALIZĖS) DALIS

### 1. Manipulioriaus valdymas

Manipuliorius – tai labiausiai paplitęs pramoninis robotas, kuris dar vadinamas robotine ranka, robotu [12] (1 pav.). Roboto ranka turi bent dvi lanksčiai sujungtas svirtis ir griebtuvą. Kiekviena svirtis turi individualius žingsninius variklius (didesni manipulioriai valdomi hidraulika ar pneumatika), kurių darbą kontroliuoja PLV arba kompiuteris.



1 pav. Medienos rąstų krovimo manipulioriaus modelis

Išmanūs manipulioriai, gebantys patys koordinuoti savo veiksmus, yra plačiai taikomi daugelyje šiuolaikinės gamybos, mokslo sričių bei įvairiose TIS srityje, tačiau robotinės rankos judėjimą apsunkina darbinėje erdvėje atsiradusios kliūtys [13]. Siekiant padidinti robotų efektyvumą ir pritaikomumą būtina sukurti išmaniąją valdymo sistemą, kurią taikant atsirastų galimybė dirbti nenumatytomis aplinkybėmis ir besikeičiančiai aplinkai. [14] Manipulioriaus išmanioji valdymo sistema taip pat gali būti taikoma kaip pagalbinė sistema operatoriui valdyti manipuliorių.

Žmonių gebėjimas judinti ranką išvengiant kliūčių ir sėkmingai pasiekti tikslą yra intuityvus. Parašyti programos algoritmą skirtą manipulioriaus valdymui, kuris atspindėtų šį žmogaus įgūdį, nėra lengva. [15] Išmanioji valdymo sistemą aprašoma kaip miniatiūrinis įtaisas, kuris apima jutimo, paleidimo ir kontrolės funkcijas. Tokia sistema pagal išanalizuotą aplinkos būseną priima atitinkamus veiksmus ir yra autonominė. Pagrindiniai taikymo uždaviniai: uždaro ciklo kontrolė, energijos efektyvumas, interneto srautų skirstymas ir t.t. Pagrindiniai išmaniosios valdymo sistemos komponentai:

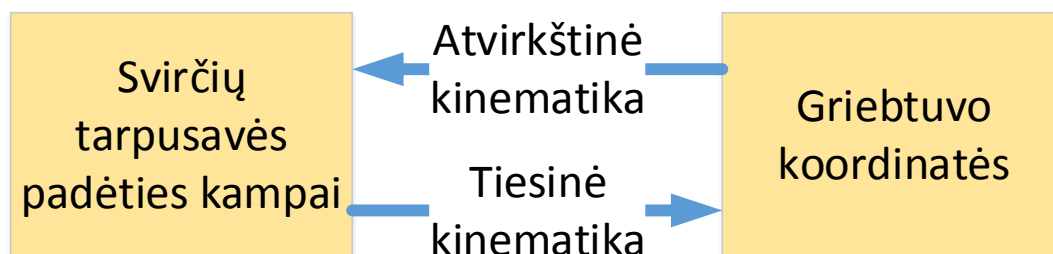
- jutikliai signalų perdavimui;
- elementai perduodantys informaciją komandų ir kontrolės blokui;
- komandų ir kontrolės blokas, kuris priima sprendimus ir pateikia nurodymus remiantis turima informacija;
- komponentai perduodantys sprendimus ir nurodymus;
- pavaros, kurios atlieka arba aktyvuoja veiksmus.

## 2. Manipulatoriaus svirčių kinematinė analizė

Robotų manipuliatorių kūrimo srityje prieš diegiant informacinės valdymo sistemas būtina išanalizuoti techninę įrangą, atlikti kinematinis skaičiavimus. Atliekant analizę susiduriama su viena iš svarbiausių problemų, kuri yra siejama su atvirkštinės kinematikos (*angl. inverse kinematics*) užduotimi [16] – lanksčiai sujungtų svirčių kampų skaičiavimu, norint tinkamai valdyti svirčių padėtį erdvėje pagal nustatytas darbo įrankio koordinates ir jo pozicionavimo sąlygas.

Atvirkštinės kinematikos problematika nagrinėjama jau daugelį dešimtmečių [17], tačiau iki šių dienų dar sprendžiama ir tampa vis labiau aktuali dėl robotų, įvairių kompiuterinių modelių populiarėjimo. Skaičiavimai susiję su atvirkštine kinematika atliekami analizuojant manipulatoriaus svirčių tarpusavės padėties kampus pagal griebtuvo poziciją koordinačių sistemoje.

Tiesinė kinematika (*angl. forward kinematics*) susijusi su kinematinėmis lygtimis, kurias taikant yra apskaičiuojama manipulatoriaus griebtuvo pozicija erdvėje žinant lanksčiai sujungtų svirčių savitarpės padėties kampų reikšmes. [11] Tiesinė kinematika neatsiejama nuo atvirkštinės kinematikos (2 pav.). Kinematiniai skaičiavimai yra naudojami robotų, manipuliatorių, animacinių filmų bei kompiuterinių žaidimų kūrimo.



2 pav. Tiesinės ir atvirkštinės kinematikos ryšiai

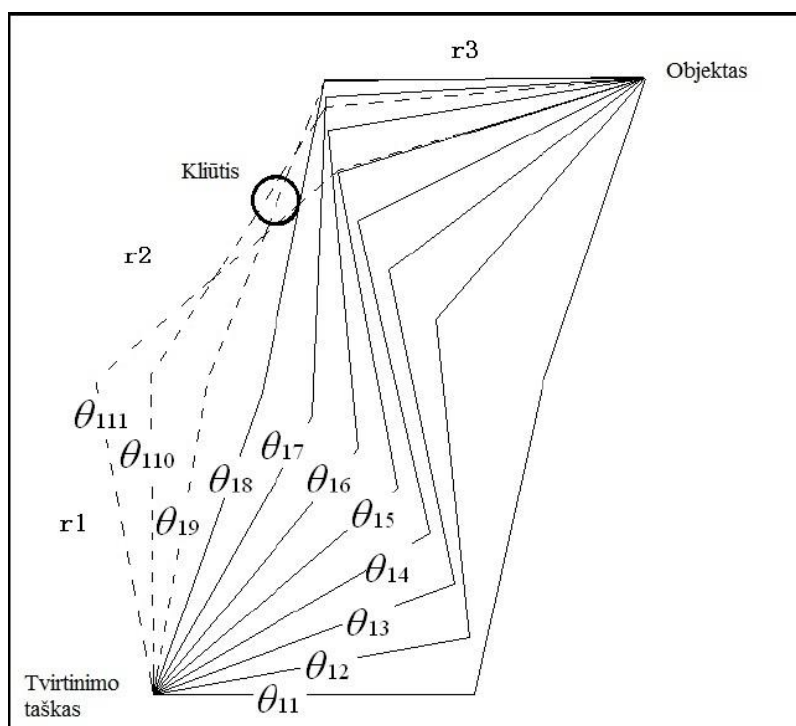
Laisvės laipsniai DOF (*angl. degrees of freedom*) – tai skaičius nepriklausomų kintamųjų, vienareikšmiškai nusakančių sistemos padėtį erdvėje. Manipulatoriaus turinčio 2 svirtis (2 laisvės laipsniai) judesiai yra apriboti: užduočiai įvykdyti tėra tik vienas variantas svirčių padėčių ir jei manipulatoriaus kelyje atsirastų kliūtis, užduotis nebūtų įvykdyta. Esant daugiau nei 2 svirtims

manipulatoriaus yra manevringesnis ir užduočiai atlikti svirčių kampinių padėčių variantų yra begalybė [18].

## 2.1. Atvirkštinė kinematika

Manipuliatorių kūrimo procese aktualią atvirkštinės kinematikos problematiką sprendžia daugybe mokslininkų, tokių kaip FU Rong, Samer Yahya ir kt.

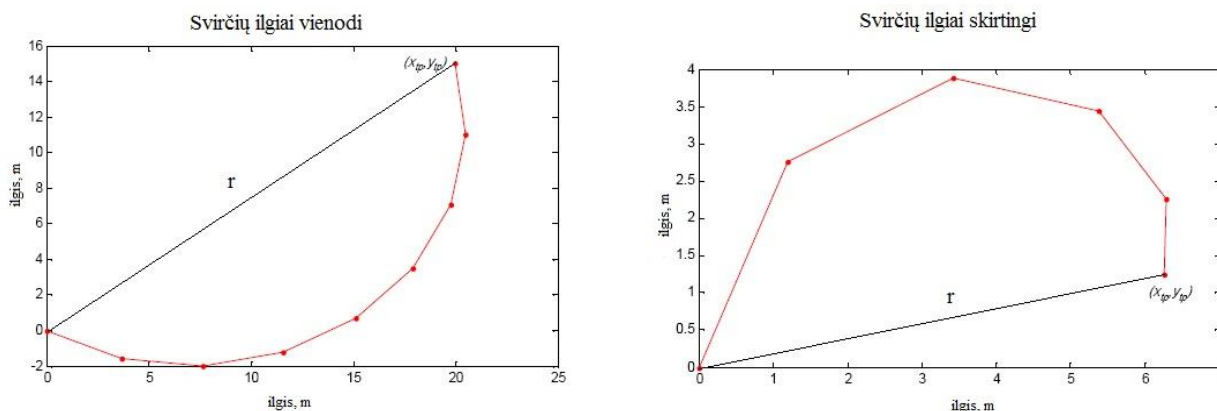
FU Rong atliko manipulatoriaus svirčių tarpusavės padėties kampų skaičiavimus, esant trims svirtims. Šiuo atveju gali būti sudaryta begalybė kampų konfigūracijų (3 pav.). Panaudojus gradiento projekcijos metodą, kuris yra paremtas Jacobian pseudo-atvirkštine lokalsios vietos optimizacija [5], randamos optimaliausios kampų padėtys užduočiai atlikti. Taikant gradiento projekcijos metodą, atsiranda galimybė išvengti sistemos veikimo zonoje atsiradusių kliūčių. Tačiau, Fu Rong pateiktas metodas nėra taikytinas trijų dimensijų erdvėje, kitaip tariant svirčių padėtys aprašomos plokštumoje, nes, anot autoriaus, trijų dimensijų erdvėje šio metodo naudojami skaičiavimai taptų per daug sudėtingi.



3 pav. Svirčių padėtys taikant gradiento projekcijos metodą [5]

Samer Yahya pateikė savo atvirkštinės kinematikos problemos sprendimo būdą. Šio metodo esmė yra suvienodinti visus manipulatoriaus svirčių tarpusavės padėties kampus, esant įvairiems svirčių ilgiams (4 pav.). Taigi, yra neįmanoma dviem ar daugiau svirčių ašių atsirasti toje pačioje vietoje [6]. Pagal atstumo tarp dviejų koordinačių plokštumoje esančių taškų skaičiavimo formulę randamas ilgis  $r$ , kurio pagalba geometriškai yra randami svirčių savitarpės padėties kampai. Naudojant šį būdą, tėra galimi du užduoties įvykdymo (svirčių padėties parinkimo) variantai:

manipuliacijos objektą pasiekia iš viršaus arba iš apačios. Dėl nesudėtingo svirčių padėčių parinkimo yra sutaupoma laiko. Tradiciniai skaičiavimo metodai matematiškai tampa pernelyg sudėtingi iš daugybės variantų išrinkti optimaliausią [7]. Tačiau, naudojant Samer Yahya metodą gali atsirasti manipuliacijos neveiksnumo zonos.



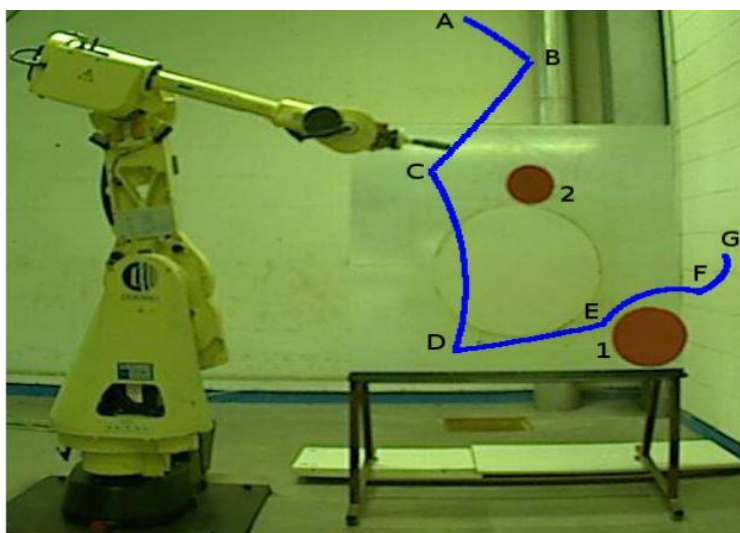
4 pav. Manipuliacijos uždavio sprendimo būdai [6]

### 3. Manipuliacijos griebtuvo trajektorija

#### 3.1. Manipuliacijos griebtuvo trajektorijos planavimas

Viena iš svarbiausių manipuliacijos išmaniosios valdymo sistemos uždavio – tinkamos trajektorijos parinkimas uždavio atlikti ir sutaupyti energetinių kaštų. Mokslininkė Francesca Ballan, ir kt. nagrinėja trajektorijos planavimą ir parinkimą.

Mokslininkė Francesca Ballan su kolegomis pateikė trajektorijos planavimo principus esant kliūtims ir pritaikė skaičiavimus manipuliatoriui COMAU SMART3-S2 (5 pav.). Sistemos privalumai – skaičiavimai supaprastinti, tad jie atliekami ypač greitai ir gali būti per teikti nuotoliniu būdu per interneto ryšį [29]. Trūkumai – dažnai pateikiamas per daug ilga griebtuvo judėjimo trajektorija, dėl to nukenčia sistemos greitis ir papildomai naudojami energetiniai resursai.

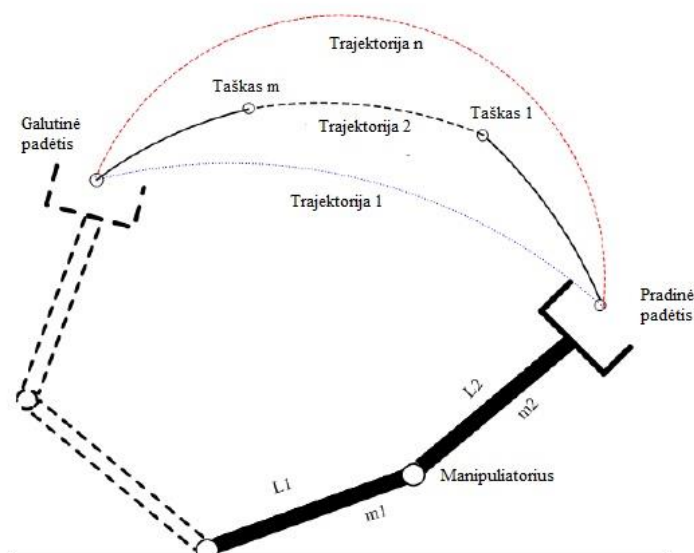


5 pav. COMAU SMART3-S2 manipuliatorius [29]

### 3.2. Manipulatoriaus griebtuvo trajektorijos optimizavimas

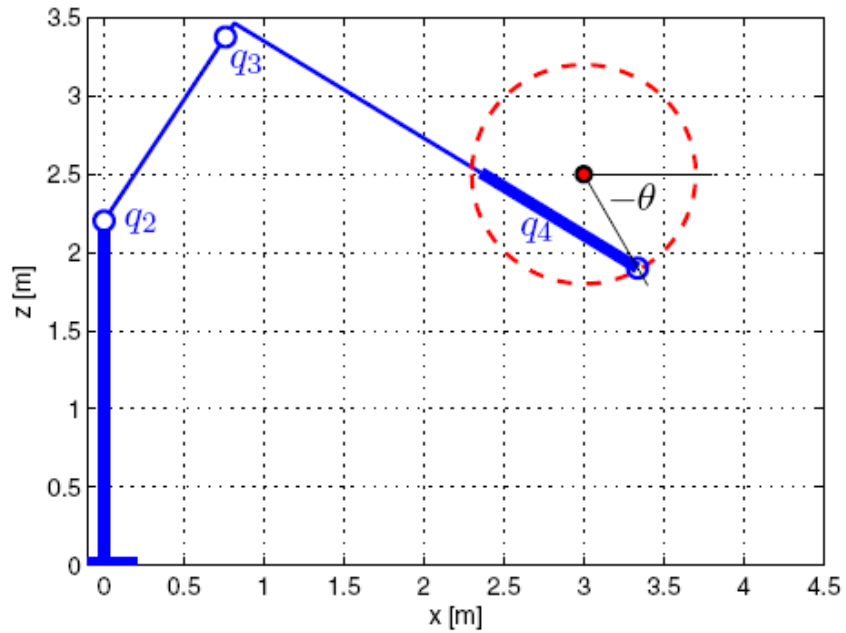
Taipogi labai svarbi problema yra susijusi su manipulatoriaus griebtuvo optimalios trajektorijos parinkimu, nuo kurio priklauso energetiškai efektyvus manipulatoriaus variklių darbas bei užduoties atlikimo trukmė [3]. Mokslininkai Zhengxiong Liu, Uwe Mettin ir kt. nagrinėja manipulatoriaus griebtuvo optimalios trajektorijos parinkimo algoritmus, siekiant minimizuoti sistemos veikimo trukmę bei energetines sąnaudas.

Zhengxiong Liu pateikia trajektorijos optimizavimo algoritmą, kurio tikslas - sumažinti užduoties įvykdymo laiką bei variklių energetines sąnaudas. Manipulatoriaus griebtuvo judėjimo trajektorijos (6 pav.). Įvedus manipulatoriaus svirčių skaičių bei griebtuvo koordinatas pradines ir objekto koordinatas, algoritmas sudaro tam tikrą trajektorijų variantų sąrašą ir tikriną trajektorijas pagal numatytus optimizavimo kriterijus. Algoritmo pagrindinis trūkumas – daug laiko užimantis optimalios trajektorijos parinkimas iš sudaryto sąrašo dėl sudėtingų optimizavimo kriterijų skaičiavimo.



6 pav. Griebtuvo judėjimo trajektorijos [9]

Uwe Mettin aptaria manipulatoriaus užduoties atlikimo trukmės minimizavimą. Mokslininkas nagrinėja teleskopinį (kintantis vienos svirties ilgis) manipuliatorių (7 pav.). Šiuo atveju įrenginys turi ne vieną sprendimo variantą užduočiai atlikti. Varikliai veikia maksimaliu greičiu bei yra paleidžiami vienu metu, todėl esant papildomam teleskopo išleidimo varikliui, galima pasiekti didesnės proceso vykdymo greitaveikos. Pagrindiniai nagrinėjamos sistemos trūkumai: dėl nesuderinto variklių darbo yra sutrumpinamas sistemos eksploatacijos laikas, o tuo pačiu gali atsirasti griebtuvo svyravimai, kurie mažina pozicionavimo tikslumą. Esant tik dviem svirtims nėra įmanoma sudaryti algoritmo kliūčių išvengimui.

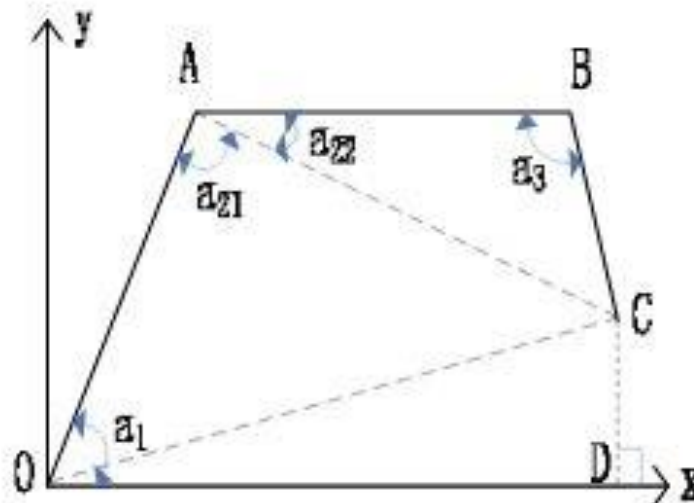


7 pav. Teleskopinis manipulatorius [19]

#### 4. Kliūčių išvengimas

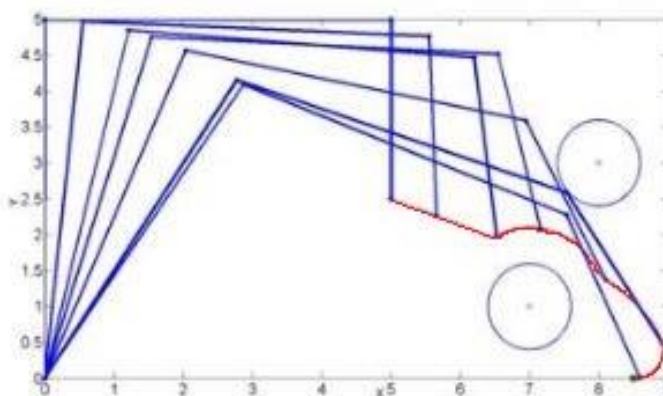
Autonominių manipuliatorių kūrime viena iš pačių svarbiausių problemų - algoritmo sukūrimas, padėsiantis išvengti manipulatoriaus svirčių judėjimo trajektorijose atsiradusių kliūčių. Mokslininkai XI Lei-ping, M. Benzaoui ir k.t. dirba šioje srityje.

Mokslininkas XI Lei-ping su kolegomis pateikia trijų svirčių manipulatoriaus (8 pav.) kliūčių išvengimo algoritmą. Manipulatoriaus svirčių tarpusavės padėties kampai yra išreiškiami per svirties ilgius. Svirčių kampinių padėčių yra daugybe variantų, bandant išvengti kliūties yra parenkamas atsitiktinai vienas iš variantų, kuris nekliudo kliūties. Sistemos trūkumas – tai atsitiktinai parenkamas svirčių kampinių padėčių variantas, dėl kurio manipulatorius neatlieka užduoties greičiausiu ir efektyviausiu būdu.



8 pav. Trijų svirčių manipulatoriaus eskizas [25]

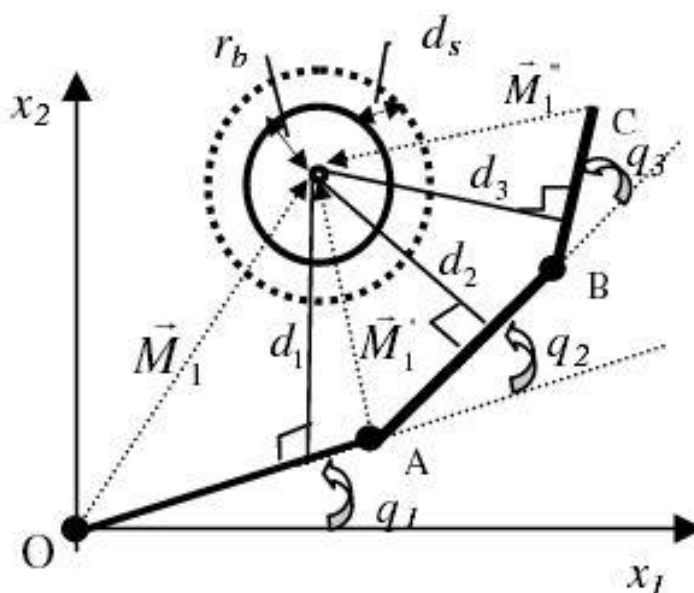
Iš mokslininko pateiktos simuliacijos (9 pav.) galime teigti, jog manipulatorius sėkmingai išvengia kliūtis. Tačiau taip pat galime pastebėti, kad raudona kreivė pavaizduoto griebtuvo judėjimo trajektorija nėra optimaliausio ilgio.



9 pav. Kliūčių išvengimo trajektorija [25]

Mokslininko XI Lei-ping pateiktas manipulatoriaus kliūčių išvengimo algoritmas sėkmingai išvengia kliūčių, tačiau metodas galėtų būti patobulintas, siekiant apdidinti variklių energetiškai efektyvų darbą ieškant trumpiausios manipulatoriaus griebtuvo judėjimo trajektorijos bei, tai atlikus, taip pat būtų sparčiau atliekama užduotis.

Mokslininkai M. Benzaoui su kolegomis analizuoja manipulatoriaus valdymą siekiant išvengti kliūčių. Pateikto metodo esmė – tai rasti kliūtis atstumą iki kiekvienos svirties. Manipulatoriui vykdant užduotį kinta atstumai nuo kliūtis iki svirties. Jei atstumas iki vienos kliūtis mažėja, stengiamasi šios svirties valdymo įtaisą sustabdyti ir užduotį atliekant valdant tik kitas svirtis. Jei atstumas nuo kliūtis iki svirties tampa pavojingai mažas, tai ta svirtis pakreipiama taip, kad atstumas iki kliūtis pradėtų didėti.



10 pav. Kliūtis atstumas iki manipulatoriaus [26]

Mokslininkui pateikus simuliaciją šio kliūčių išvengimo metodo įrodoma, jog šis sudėtingas skaičiavimų algoritmas veikia, tačiau visgi pasitaiko 2 – 3% klaidų ir kliūtis yra kliudoma arba manipulatorius per ilgai atlieka skaičiavimus ir kampų posūkius, bet užduoties nepasiekia nekliudydamas kliūtis.

## II. METODINĖ DALIS

### 1. Manipulatoriaus kūrimo problematika

Įvairiuose gamybos bei technologiniuose procesuose reikalingos sistemos sparčiai atliekančios paskirtas užduotis [21]. Analizuotos kitų mokslininkų manipulatoriaus valdymo sistemos nepasižymi ypatinga greitaveika dėl sudėtingų manipulatoriaus svirčių savitarpės padėties kampų bei trajektorijos parinkimų skaičiavimų. Todėl būtina atlikti alternatyvius skaičiavimus bei sudaryti jais grindžiamus algoritmus, kurie būtų matematiškai nesudėtingi (valdiklis sparčiai atliktų skaičiavimus), tačiau manipulatorius galėtų pilnai funkcionuoti ir vykdyti paskirtas užduotis.

Tiriamai TIS ypatingai reikalingas energetiškai efektyvus variklių darbas, siekiant sumažinti eksploatacijos kaštus bei atitinkamai platesnio manipulatoriaus pritaikymo, pvz.: kosmose, povandeninėje aplinkoje, taipogi įvairiose medicinos srityse bei mokslinių tyrimų taikyme. Įvertinus aplinkos specifika, ten naudojami elektriniai varikliai, nes deguonies stygius gali sutrikdyti normalų vidaus degimo variklių darbą. Tam tikrais atvejais, kai energija netiekama iš papildomų atsinaujinančių šaltinių, tokių kaip saulės jėgainės, būtina numatyti papildomus maitinimo elementus. Tačiau, pastarieji kol kas neturi didelės talpos, todėl siekiant optimizuoti variklių darbą būtina sudaryti skirtingoms situacijoms pritaikytus optimalios trajektorijos parinkimo algoritmus.

Tiriama manipulatoriaus informacinė valdymo sistema nėra taikoma konkrečioje srityje. Tačiau, nepriklausomai nuo srities iškyla kliūčių aptikimo bei jų išvengimo problema, kuriai išspręsti reikalinga išmanioji sistema nustatanti užduoties ir kliūtis pozicijas, įvertinanti aplinkos sąlygas ir pagal jas keičianti proceso eigą bei didelis dėmesys turi būti skiriamas manipulatoriaus darbo teritorijoje dirbančio ar pašalinio žmogaus sveikatai [22].

Ypač produktyvių techninių IS poreikis apibrėžiamas aukšto tinkamumo ir padidinto funkcionalumo bei automatizavimo lygmens reikalingumu. Orientuojantis į įvairius pramoninius ar kt. procesus, siekiamas aukščiausias automatizavimo lygmuo – valdymo sistemai paskirtų užduočių atlikimas be visiško žmogaus įsikišimo. Šiuo atveju sistema turi pati priimti sprendimus, atsižvelgiant į aplinkos sąlygas ir susiklosčiusią proceso metu situaciją, tuo pačiu padaryti prielaidas ir atitinkamai reaguoti. Tokiu būdu galima pasiekti sistemos našumo bei išvengti nepageidautinų įvykių.

Išsprendus nagrinėjamas problemas (optimizuojant variklių darbus, parenkant optimalią manipulatoriaus griebtuvo judėjimo trajektoriją ir išvengiant manipulatoriaus judėjimo trajektorijoje esančių kliūčių) taip pat būtų padidintas sistemos veikimo našumas ir saugumas. Atlikus alternatyvius, valdiklio matematiškai lengviau ir greičiau apdorojamus, manipulatoriaus svirčių kampinės padėties skaičiavimus, būtų galima sutrumpinti užduoties atlikimo trukmę. Manipulatoriaus valdymo sistemai sudarius specialų griebtuvo trajektorijos parinkimo algoritmą, atsiranda galimybė sumažinti variklių energijos sąnaudas [23], kas labai svarbu sprendžiant

finansines, bei ekologines problemas. Taipogi pagal ši algoritmą parinkta trajektorija leidžia saugiai atlikti užduotis išvengiant kliūčių, kas ypač svarbu siekiant įgyvendinti autonominio manipulatoriaus eksploatavimą.

Sukurtas manipulatoriaus valdymo sistemos virtualus modelis leidžia imituoti realiai sistemai adekvačius technologinius procesus. Kiekviename sistemos kūrimo etape gauti rezultatai, atlikus atitinkamus skaičiavimus ir panašių sistemų palyginimus, leidžia suprojektuoti realios sistemos modelį ir padeda tikslingai parinkti technines dalis bei sudaryti optimalius algoritmus.

## 2. Manipulatoriaus kūrimo sprendimas

Šiame darbe nagrinėjamas trijų svirčių manipulatoriaus realaus laiko valdymo sistema. Svirčių ilgiai parinkti vienodi, mes tokie manipulatoriai yra universalūs ir gali būti plačiai taikomi. Tačiau šiai sistemai taip pat sėkmingai gali būti parenkami ir skirtingų ilgių svirtys. Manipulatoriuje taikomi žingsniniai varikliai dėl ypatingai tikslaus jų veikimo ir plataus pritaikomumo. Šie elektriniai varikliai gali būti taikomi kosmose ar po vandeniu, kur nėra deguonies, kur vidaus degimo varikliai negalėtų būti naudojama. Aišku, tiek hidraulika, tiek pneumatika gali veikti su elektriniai tepalo siurbliais, tačiau tokiu atveju reikalinga nemažai papildomų jutiklių galinčių atpažinti svirčių tarpusaves padėtis ir šiuo atveju nebūtų galima pasiekti precizinio tikslumo. Norint sukurti manipulatoriaus valdymo sistemą, būtina išanalizuoti žingsninių variklių valdymo principus, pagal kuriuos būtų galima parašyti valdymo algoritmą. Manipulatoriaus užduoti koordinačių ašyse apibrėžia operatorius arba užduoties padėtis nuskaitoma jutiklių pagalba – tai priklauso nuo to, kokioje aplinkoje manipulatorius yra taikomas. Kai manipulatorius yra stacionarioje būsenoje ir užduoties koordinatės keičiasi nežymiai – koordinatės nuskaitomos jutikliais. Kai užduoties apimtis yra nuolat kintanti – operatorius nustato kurioje vietoje ir kokio dydžio yra užduotis, pvz. iš keleto rąstų manipulatoriui reikia paimti tik vieną ir pan. Kliūtis pozicija nustatymui naudojama 2 ar daugiau ultragarsinių jutiklių, kurių duomenis apdorojus nustatomos tikslios kliūtis koordinatės. Šiame darbe nėra detalai analizuojamas objektų pozicijos radimas – sprendžiamos pagrindinės manipulatoriaus išmaniosios sistemos kūrimo problemos: kinematiniai skaičiavimai, trajektorijos parinkimas ir optimizavimas bei kliūčių išvengimas. Kaip ir daugelyje mokslinių publikacijų skaičiavimai yra pateikti dviejų dimensijų erdvėje. Pateiktas manipulatorius trečioje dimensijoje ( $z$  ašyje) turi tik vieną laisvės laipsnį –  $z$  ašyje sukiojamas vienu varikliu, įtaisytu tarp manipulatoriaus ir pagrindo. Esant vienam laisvės laipsniui skaičiavimai nėra sudėtingi, todėl šiuo atveju nagrinėjama tik dviejų dimensijų ašyje.

### 3. Žingsninių variklių analizė

#### 3.1. Žingsniniai varikliai ir jų parametrai

Žingsninis variklis – tai elektros mašina, pakeičianti valdymo impulsus į kampinį arba linijinį poslinkį. Dėl lengvo pritaikomumo, tikslumo ir mažo dydžio, žingsniniai varikliai yra plačiai taikomi nedidelėse automatinėse sistemose [24]. Šie varikliai, kartu su komutatoriumi, naudojami kaip vykdymo įtaisai, kuriems nereikia taikyti padėties jutiklių. Tokie vykdymo įtaisai priskirtini dažniu valdomiems sinchroniniams varikliams. Įvairūs varikliai skiriasi fazių skaičiumi (nuo 2 iki 6) ir rotoriaus konstrukcija. Rotorius gali būti aktyvus (žadinamas pastoviais magnetais) ir pasyvus (reaktyvusis). Varikliai su aktyviu rotoriumi turi didesnę galią, bet mažesnę tikslumą (didesnį žingsnį). Žingsniu vadinamas kampas, kuriuo pasisuka rotorius po vieno impulso. Variklių su pasyviu rotoriumi žingsnis būna  $1^\circ - 9^\circ$ , o su aktyviu –  $10^\circ$  ir daugiau. Nuo impulsų padavimo eiliškumo priklauso variklio sukimosi kryptis.

Norint išvengti žingsninių variklių veikimo netikslumų, būtina žinoti charakteristikas apribojančias dinامينius procesus:

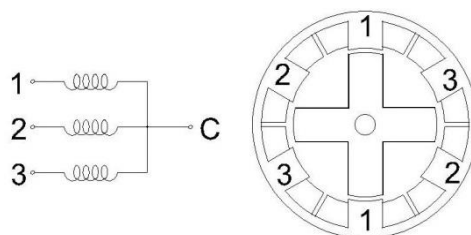
*Paleidimo momentas* – didžiausias galimas sukimo momentas varikliui dar nepradėjus sukis.

*Maksimalus variklio paleidimo dažnis* – tai dažnis, kuriuo, komutuojant variklio apvijas, galima priversti sukis variklio rotorius nepraleidžiant nė vieno takto. Esminę įtaką šiam dažniui daro rotoriaus ir apkrovos inercijos momentai.

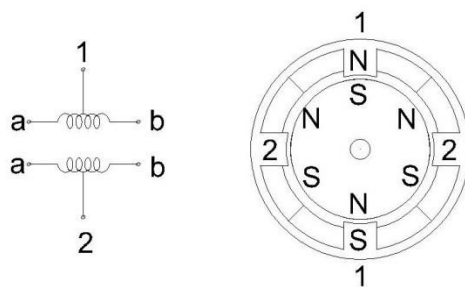
*Maksimalus komutavimo dažnis* – viršijus šį dažnį, srovė apvijoje per vieną taktą jau nebesiekia savo nominalios vertės. Komutavimo dažnį apriboja apvijų induktyviosios varžos didumas ir srovės forsavimo schema.

Dažniausiai naudojamų žingsninių variklių tipai:

- Vieno poliaus žingsninis variklis gali būti kelių rūšių. Vienu atveju variklis turi tris apvijas, sujungtas taške C bei tris variklio valdymo išvadus (11 pav.). Kitu atveju (12 pav.) žingsninis variklis turi penkis pajungimui skirtus išvadus (1 ir 2 laidai sujungiami kartu) ir dvi variklio apvijas, kurios būna atskiruose korpusuose. Tokio tipo variklis valdomas stačiakampiais amplitudės impulsais, atitinkama seka paduodamais į variklio apvijas.

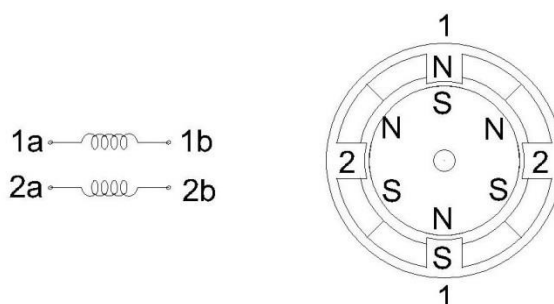


11 pav. Žingsninis variklis turintis tris išvadus



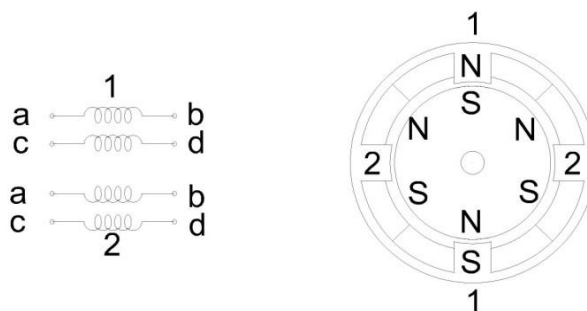
12 pav. Žingsninio variklio apvijų atskiruose korpusuose

- Dviejų polių žingsninis variklis turi keturis išvadus bei dvi apvijas (13 pav.). Valdomas taip pat stačiakampiais impulsais atitinkama seka paduodamais į variklio apvijas, tačiau su kiekvienu nauju impulsu yra pakeičiamas jo poliariškumas.



13 pav. Dviejų polių žingsninis variklis

- Daugiapolis žingsninis variklis (14 pav.) turintis keturias apvijas ir aštuonis išvadus. Toks variklis valdomas stačiakampiais impulsais.



14 pav. Daugiapolis žingsninis variklis

### 3.2. Valdymo pobūdis

Žingsniniams varikliams valdyti naudojami mikrovaldikliai, nes jais yra paprasta generuoti įvairaus dažnio ir trukmės impulsus. Mikrovaldikliai pasižymi mažomis elektros energijos sąnaudomis, santykinai dideliu greičiu ir yra ganėtinai pigūs. Atitinkamai užprogramuotas mikrovaldiklis gali dirbti kaip skaitiklis ar bet koks kitas prietaisas, kuriam sukonstruoti reikėtų daug paprastų mikroschemų, tuo tarpu mikrovaldikliai yra vienos paprasčiausios mikroschemos gabaritų.

Vieno ir dviejų polių žingsniniai varikliai yra valdomi keturių išėjimų mikrovaldikliais, tačiau skirtingais impulsų padavimo principais. Vieno polio varikliuose impulsai turi būti paduodamas eilės tvarka ir vienu metu tik po vieną impulsą, siekiant varikliui pasisukti pilnu žingsniu (Lentelė 1). Šie varikliai taip pat gali dirbti sustiprintos galios režimu arba pasisukti per pusę žingsnio (žr. 3 priedą). Pusės pasisukimo principą žiūrėti 4 priede.

Lentelė 1. Pilno žingsnio impulsų seka

Impulsas	Apvija a1	Apvija b1	Apvija a2	Apvija b2
1	+			
2		+		
3			+	
4				+

Dviejų polių žingsninis variklis valdomas panašiai kaip vieno, tačiau šiame variklyje kas kiekvieną impulsą yra keičiamas jo poliariškumas. Vieno poliaus impulsų padavimo laiduose poliariškumas nekinta ir pagal pasirinkimą yra teigiamas arba neigiamas. Dviejų polių žingsninių variklių išvaduose vienu metu impulsas yra teigiamas, kitu metu neigiamas poliariškumo.

Lentelė 2. Dviejų polių variklio impulsai

Impulsas	Apvija a1	Apvija b1	Apvija a2	Apvija b2
1	+		+	
2		+	+	
3		+		+
4	+			+

2 lentelėje pavaizduoti dviejų polių žingsninių variklių padavimo impulsai. Paduodama po du impulsus, todėl tai yra padidintos galios variklio veikimas: vienos apvijos sudarytas magnetinis laukas stumia rotorį, kitos apvijos – traukia. Kaip minėta, dviejų polių varikliuose paduodamas ir teigiamas ir neigiamas impulsas. Šioje lentelėje pavaizduotas tik teigiamas impulsas, tačiau jis vieną apvijos galą impulsas pasiekia tiesiogiai, o kitą per *npn* tranzistorių, kuris teigiamą impulsą pakeičia neigiamu.

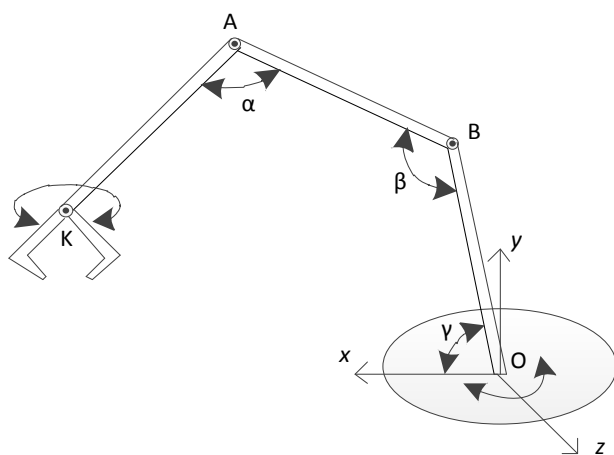
### III. PROJEKVINĖ DALIS

#### 1. Atvirkštinės kinematikos sprendimas

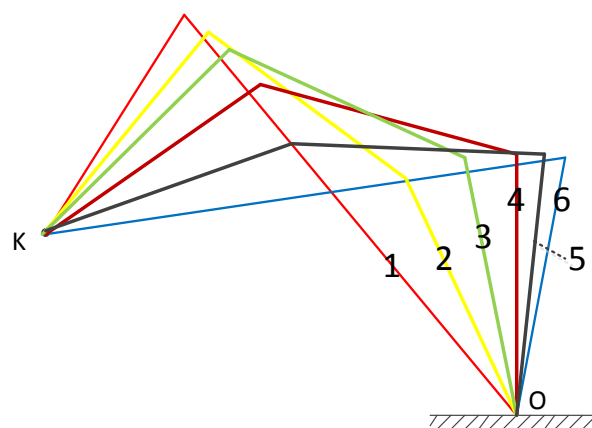
##### 1.1. Manipulatoriaus veikimo principas

Šiame darbe nagrinėjamas trijų svirčių manipulatoriaus modelis (15 pav.). Esant trimis svirtims ir daugiau gali būti sudaryta begalybė galimų svirčių kampų konfigūracijų užduočiai atlikti. [27] Sekančiame paveiksle (16 pav.) pavaizduota keletas svirčių padėčių variantų manipulatoriaus griebtuvui esant taške  $K$ . Siekiant sumažinti variklių energetines sąnaudas, labai svarbu parinkti optimaliausias manipulatoriaus svirčių tarpusavės padėtis.

Manipulatoriaus valdymo sistema įsimena griebtuvo ( $K$  taško) koordinatas ir svirčių kampines padėtis po paskutinės užduoties atlikimo. Pagal užduotas naujas objekto, kurį turi pasiekti griebtuvas, koordinatas sistema apskaičiuoja koku kampu atitinkamai pasukti kiekvieną svirtį nuo savo pradinių padėčių, kad būtų minimizuoti energetiniai kaštai: visų variklių bendras energijos suvartojimas būtų minimalus. [28]



15 pav. Manipulatoriaus modelis



16 pav. Svirčių tarpusavės padėties variantai

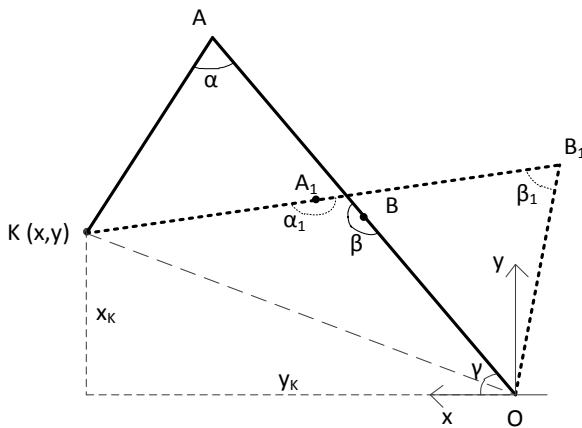
Taigi, siekiant minimizuoti variklių energetinius kaštus, yra būtina sudaryti lygtis, pagal taško  $K$  koordinatas apskaičiuojančias svirčių tarpusavės padėties kampų konfigūracijas. Konfigūracijų kiekis apribojamas įvedus kampo posūkio žingsnį, nagrinėjamu atveju  $0,01^0$ . Manipulatoriaus svirčių judesių koordinavimas aprašomas kampais  $\alpha$ ,  $\beta$ ,  $\gamma$  (15 pav.) ir jų vertės yra intervale  $0^0 - 180^0$ .

Žinant objekto koordinatas yra paskaičiuojami visi įmanomi manipulatoriaus svirčių padėties kampai  $0,01^0$  žingsniu, kurie suvedami į lentelę. „C++“ programos pagalba lentelėje esantys kampai lyginami su pradinės padėties kampais ir parenkamas mažiausiai energetinių resursų naudojantis variantas griebtuvui pasiekti objektą.

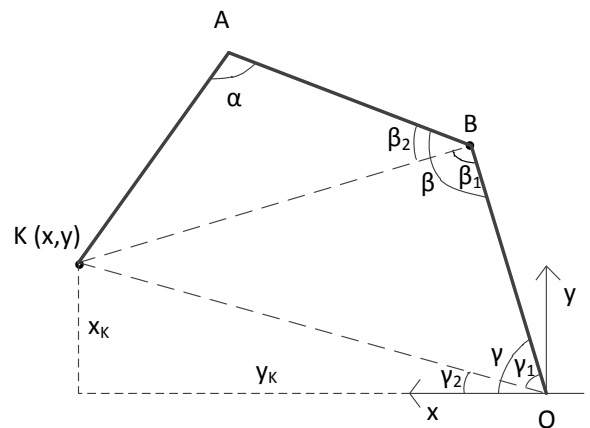
## 1.2. Svirčių kampinės padėties skaičiavimas

Iš pradžių randamas pasirinkto kampo (šiuo atveju -  $\alpha$ ) intervalas pagal objekto koordinates. Sudaroma kampo  $\alpha$  aibė, kurioje yra visi kampai nuo minimalios  $\alpha$  padėties iki maksimalios besiskiriantys žingsniu –  $0,01^0$ . Išvedamos  $\beta$ ,  $\gamma$  kampo skaičiavimo formulės, kuriose vertės paskaičiuojamos prie kiekvienos  $\alpha$  aibės reikšmės.

Svirtys yra vienodo ilgio, todėl  $KA = AB = BO = OB_1 = B_1A_1 = A_1K = L$ . Objektas negali būti toliau negu  $3L$  (pilnai ištiestos manipulatoriaus svirtys) ir ne arčiau nei per vieną svirtį ( $KO > L$ ) nuo manipulatoriaus įtvirtinimo prie pagrindo. Kampo  $\alpha$  intervalai skaičiuojami dviem būdais, priklausomai nuo taško  $K$  koordinačių.



17 pav. Manipulatoriaus ribiniai kampai



18 pav. Svirčių kampinės padėties

Tikriname ar  $3L > KO > L$ . Jei nelygybė teisinga, tai kampas  $\alpha$  gali įgyti mažiausią įmanomą vertę, kai svirtys sudarys trikampį  $AOK$ , tada  $\beta$  įgis maksimalią vertę -  $180^0$ . Pagal formulę (1) randamas  $KO$ , o  $L$  yra žinomas (svirčių ilgis). Kampo  $\alpha$  minimali vertė skaičiuojama naudojant kosinusų teorema (17 pav. **Error! Reference source not found.**):

$$KO = \sqrt{x_k^2 + y_k^2}; \quad (1)$$

$$KO^2 = L^2 + (2L)^2 - 2L(2L) \cos \alpha; \quad (2)$$

$$\alpha = \cos^{-1} \left( \frac{5L^2 - (x_k^2 + y_k^2)}{4L^2} \right) \quad (3)$$

Naudojant kosinusų teoremą (17 pav.) per kampus  $\alpha$  ir  $\gamma_1$  yra išreiškiama  $KB^2$ . Sulyginus (4) ir (5) formulių dešines puses išreiškiamas kampas  $\gamma_1$  ir iš (1) formulės įstatoma  $KO$  išraiška:

$$KB^2 = L^2 + L^2 - 2L^2 \cos \alpha; \quad (4)$$

$$KB^2 = KO^2 + L^2 - 2KO \cdot L \cos \gamma_1; \quad (5)$$

$$\gamma_1 = \cos^{-1} \frac{x_k^2 + y_k^2 - L^2(1 - 2 \cos \alpha)}{2L \sqrt{x_k^2 + y_k^2}}. \quad (6)$$

Turint kampą  $\gamma_1$  paskaičiuojamas kampas  $\gamma$ :

$$\gamma = \gamma_1 + \gamma_2; \quad (7)$$

$$\gamma_2 = \tan^{-1} \frac{x_k}{y_k}; \quad (8)$$

$$\gamma = \cos^{-1} \frac{x_k^2 + y_k^2 - L^2(1 - 2 \cos \alpha)}{2L\sqrt{x_k^2 + y_k^2}} + \tan^{-1} \frac{x_k}{y_k}. \quad (9)$$

$\beta$  išreiškiamas kaip  $\beta_1$  ir  $\beta_2$  suma (17 pav.). Pagal kosinusų teoremą išreiškiamas kampas  $\beta_1$ . Trikampis  $ABK$  lygiašonis, todėl kampas  $\beta_2$  ir kampas  $AKB$  yra lygus ir išreiškiamas pagal (13) formulę. (14) formulėje sudėjus  $\beta_1$  ir  $\beta_2$  vietoj  $KB^2$  įkelta formulė (5):

$$\beta = \beta_1 + \beta_2; \quad (10)$$

$$KO^2 = KB^2 + L^2 - 2KB \cdot L \cos \beta_1; \quad (11)$$

$$\beta_1 = \cos^{-1} \frac{KB^2 + L^2 - KO^2}{2KB \cdot L}; \quad (12)$$

$$\beta_2 = \frac{180^\circ - \alpha}{2}; \quad (13)$$

$$\beta = \cos^{-1} \frac{L - KO \cos \gamma_1}{\sqrt{KO^2 + L^2 - 2KO \cdot L \cos \gamma_1}} + \frac{180^\circ - \alpha}{2}. \quad (14)$$

### 1.3. Kampai aprašantys manipulatoriaus svirčių judesių koordinavimą

Pasinaudojus (3), (9) ir (14) formulėmis galima sudaryti manipulatoriaus svirčių judesių koordinavimą aprašančių kampų  $\beta$ ,  $\gamma$  kitimą, kampui  $\alpha$  kintant  $0,01^\circ$  žingsniu. Kuo mažesnis kampo  $\alpha$  kitimas, tuo daugiau variantų algoritmas pateikia [1]. Gauti kampai pateikti lentelėje 1.

Kadangi manipulatorius turi 3 laisvės laipsnių koordinacių plokštumoje - svirčių tarpusavės padėties kampų variantų gali būti begalė. Taigi, šių skaičiavimų pagalba bei įvedus kampo  $\alpha$  kitimo didumą yra apskaičiuojami visi įmanomi variantai pagal užduotą kampo  $\alpha$  kitimo vertę.

Lentelės pirmoje eilutėje yra apskaičiuojama mažiausia įmanoma kampo  $\alpha$  užduočiai atlikti. Radus šį kampą manipulatorius tampa 2 laisvės laipsnių, todėl kampams  $\beta$ ,  $\gamma$  lieka tik vienintelis korektiškas verčių variantas. Taigi, pagal rastą kampą  $\alpha$  yra apskaičiuojami kampai  $\beta$ ,  $\gamma$ . Sekančiose eilutėse paeiliui prie kampo  $\alpha$  yra pridėjama  $0,01^\circ$  vertė ir vėl skaičiuojami kampai  $\beta$ ,  $\gamma$ . Lentelė yra pildoma kol kampas  $\alpha$  pasiekia  $180^\circ$  ribą. Paskutiniaisiais kampų variantas, kai kampas  $\alpha$  viršija  $180^\circ$  kampą lentelėje nebepildomas.

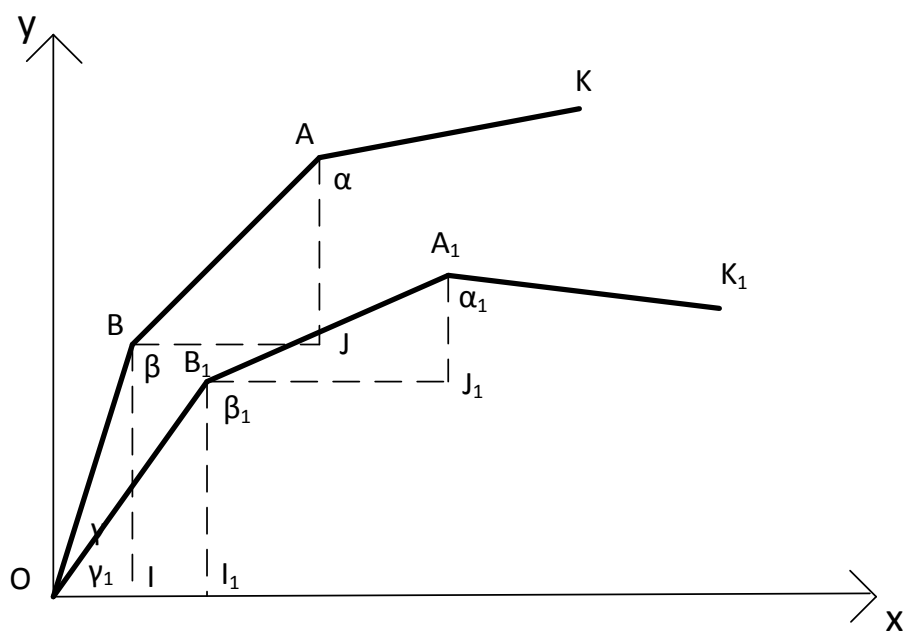
Lentelė 3. Manipulatoriaus svirčių judesių koordinavimą aprašantys kampai

Nr.	Kampas $\alpha$	Kampas $\gamma$	Kampas $\beta$
1.	$\cos^{-1} \left( \frac{5L^2 - (x_k^2 + y_k^2)}{4L^2} \right)$	$\cos^{-1} \frac{x_k^2 + y_k^2 - L^2(1 - 2 \cos \alpha)}{2L \sqrt{x_k^2 + y_k^2}} + \tan^{-1} \frac{x_k}{y_k};$	$\cos^{-1} \frac{L - KO \cos \gamma_1}{\sqrt{KO^2 + L^2 - 2KO \cdot L \cos \gamma_1}} + \frac{180^0 - \alpha}{2}$
2.	$\cos^{-1} \left( \frac{5L^2 - (x_k^2 + y_k^2)}{4L^2} + 0,01^0 \right)$	$\cos^{-1} \frac{x_k^2 + y_k^2 - L^2(1 - 2 \cos \alpha)}{2L \sqrt{x_k^2 + y_k^2}} + \tan^{-1} \frac{x_k}{y_k};$	$\cos^{-1} \frac{L - KO \cos \gamma_1}{\sqrt{KO^2 + L^2 - 2KO \cdot L \cos \gamma_1}} + \frac{180^0 - \alpha}{2}$
3.	$\cos^{-1} \left( \frac{5L^2 - (x_k^2 + y_k^2)}{4L^2} + 0,02^0 \right)$	$\cos^{-1} \frac{x_k^2 + y_k^2 - L^2(1 - 2 \cos \alpha)}{2L \sqrt{x_k^2 + y_k^2}} + \tan^{-1} \frac{x_k}{y_k};$	$\cos^{-1} \frac{L - KO \cos \gamma_1}{\sqrt{KO^2 + L^2 - 2KO \cdot L \cos \gamma_1}} + \frac{180^0 - \alpha}{2}$
4.	...	...	...
...	$180^0$	$\cos^{-1} \frac{3(x_k^2 + y_k^2 - L^2)}{2L \sqrt{x_k^2 + y_k^2}} + \tan^{-1} \frac{x_k}{y_k}$	$\cos^{-1} \frac{L - KO \cos \gamma_1}{\sqrt{KO^2 + L^2 - 2KO \cdot L \cos \gamma_1}}$

Ši sudaryta lentelė leidžia, pasinaudojus „C++“ programa, parinkti optimaliausią manipulatoriaus svirčių kampinės padėties variantą, atsižvelgiant į atitinkamų kampų reikiamą pokytį užduočiai atlikti. Manipulatoriaus pradinės būsenos kampai  $\alpha$ ,  $\beta$ ,  $\gamma$  „C++“ programoje yra lyginami su kiekviena eilute iš lentelės 1 ir yra išrenkami kampai, kurie mažiausiai skiriasi nuo pradinių kampų. Jų reikšmės – tai mažiausiai resursų reikalaujantis variantas užduočiai atlikti [2].

## 2. Tiesinės kinematikos sprendimas

Tiesinės kinematikos skaičiavimai taikomi skaičiuojant svirties pabaigos koordinatas ar griebtuvo poziciją erdvėje žinant lanksčiai sujungtų svirčių savitarpės padėties kampų reikšmes. Tolimesniems skaičiavimams reikalingos antrosios ir trečiosios svirties (jungiasi su griebtuvu) – taškas  $A$  ir  $A_I$  (19 pav.), tačiau tarpiniams skaičiavimams reikalinga rasti ir  $B$  ir  $B_I$  koordinatas.



19 pav. Tiesinės kinematikos skaičiavimai

Svirčių tarpusavės padėties kampai apskaičiuoti ankstesniame skyriuje „Atvirkštinės kinematikos sprendimas“. Jais naudojantis randamos taško  $A$  koordinatės. Prieš tai reikalinga surasti  $B$  kampo koordinatės:

$$x_B = OI = BO \cos \gamma = L \cos \gamma \quad (15)$$

$$y_B = BI = BO \sin \gamma = L \sin \gamma \quad (16)$$

Sekančiame žingsnyje randamas kampas  $ABJ$ :

$$ABJ = \beta - 90^\circ - (180^\circ - 90^\circ - \gamma) \quad (17)$$

Pagal kampą  $ABJ$  randama kokių atstumu nuo taško  $B$  yra nutolęs taškas  $A$ :

$$x_a = BJ = AB \cos ABJ = L \cos ABJ \quad (18)$$

$$y_a = AJ = AB \sin ABJ = L \sin ABJ \quad (19)$$

Turint apskaičiuotas koordinatės randamos  $A$  taško koordinatės:

$$x_A = x_B + x_a = L \cos \gamma + L \cos ABJ \quad (20)$$

$$y_A = y_B + y_a = L \sin \gamma + L \sin ABJ \quad (21)$$

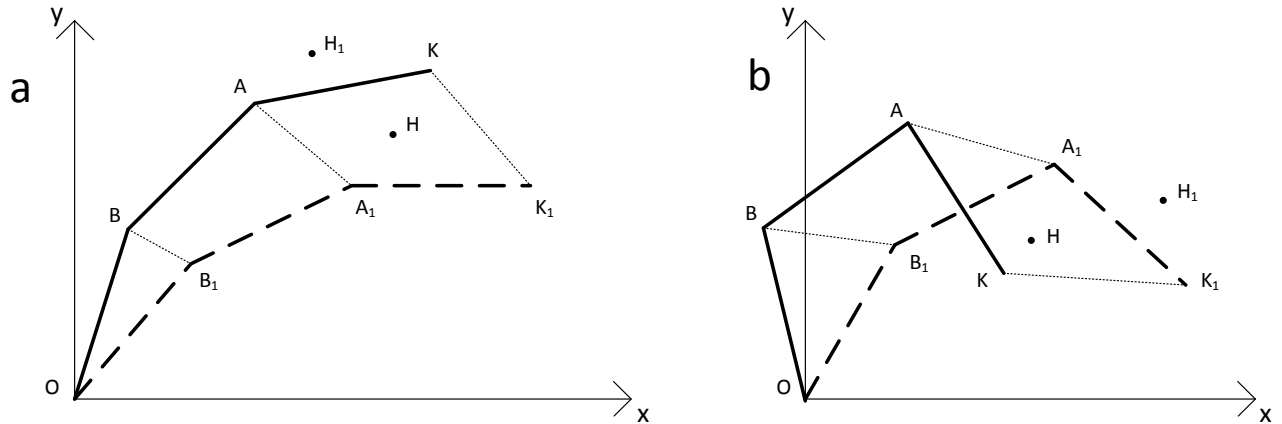
Analogiškai randamos ir taško  $A_1$  koordinatės:

$$x_{A_1} = x_{B_1} + x_{a_1} = L \cos \gamma + L \cos A_1B_1J_1 \quad (22)$$

$$y_{A_1} = y_{B_1} + y_{a_1} = L \sin \gamma + L \sin A_1B_1J_1 \quad (23)$$

### 3. Kliūčių pozicijos analizė ir išvengimo metodas

Jutikliams užfiksavus kliūtį manipulatoriaus veikimo zonoje, valdymo sistema turi patikrinti ar kliūtis trukdo atlikti užduotį ar ne, t.y. ar kliūtis patenka į manipulatoriaus svirčių judėjimo trajektorijas.



20 pav. Kliūčių pozicijos manipulatoriaus judėjimo trajektorijoje

20 pav. abiejuose variantuose a ir b tiesiomis linijomis pavaizduotos manipulatoriaus svirtys pradinėje būsenoje. Punktūrinėmis linijomis – svirčių padėtys apskaičiuotos pagal 2.1. skyrelyje aprašytą svirčių tarpusavės padėties kampų skaičiavimo pateiktą metodą, siekiant atlikti užduotį. Kliūčiai kaip ir užduoties taškui galioja ta pati sąlyga -  $3L > H > L$  ( $L$  - svirties ilgis).

Iš 20 pav. matosi manipulatoriaus svirčių brėžiamas plotas svirtims judant nuo pradinių padėčių  $OABK$  iki padėčių pasiekus užduotį –  $OA_1B_1K_1$ . Jei kliūtis nepatenka į šiuos brėžiamus plotus, pavyzdžiui kliūtis  $H_1$ , tai galima teigti, jog kliūtis nepatenka į manipulatoriaus judėjimo trajektoriją. Tokiu atveju manipulatorius valdomas įprastai – tiesiog varikliai pakeičia svirčių tarpusavės padėties kampus pagal jau apskaičiuotą variantą  $OA_1B_1K_1$ . Tačiau, jei kliūtis patenka į manipulatoriaus svirčių brėžiamą plotą (pavyzdžiui kliūtis  $H$ ), tai būtina naudoti kliūčių išvengimo metodą, kuris aprašomas sekančiame skyrelyje.

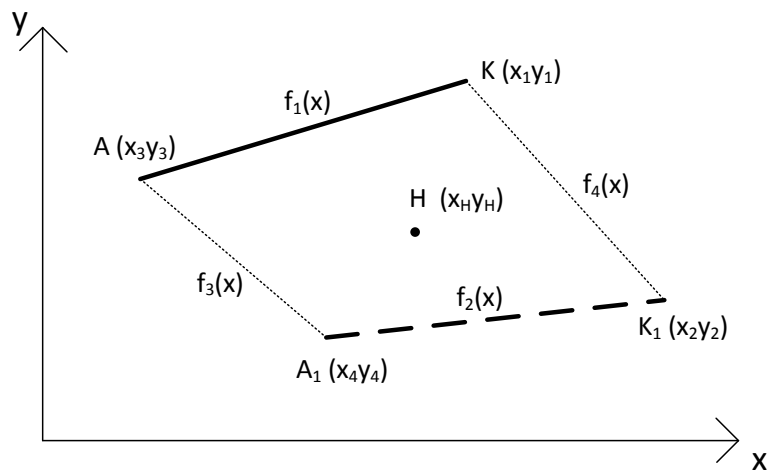
Taigi, manipulatoriaus valdymo sistema tikrina ar kliūtis yra svirčių brėžiamuose platuose  $AKA_1K_1$ ,  $ABA_1B_1$  (žiūrėti 20 pav.).  $OBB_1$  plote kliūties buvimas negalimas pagal užduotą sąlygą.

Paveiksle svirčių susijungimo taškų judėjimą apibrėžia tiesės  $BB_1$ ,  $AA_1$ ,  $KK_1$ . Realioje situacijoje tai yra kreivės, tačiau skaičiavimo patogumui atvaizduojamos kaip tiesės, nes  $BB_1$  pakeitimas tiese įtakos realiam veikimui neturi – trikampis  $OBB_1$  su  $ABA_1B_1$  susijungia per  $BB_1$  ir plotas tarp linijos ir kreivės kuriai figūrai  $OBB_1$  ar  $ABA_1B_1$  priklausys nėra svarbu. Tam tikrose situacijose  $AA_1$ ,  $KK_1$  pakeitimas į tieses gali nežymiai įtakoti tikslingą sistemos veikimą, taigi siekiant išvengti svirčių užkliuvimo už kliūtis turi būti įvedamas koeficientas, paskaičiuojamas pagal svirčių tarpusavės padėties kampų pasikeitimą nuo pradinės reikšmės iki galutinės reikalingos užduočiai atlikti. Nuo šio koeficiento priklausytų koku atstumu kliūtis turi būti nuo svirčių

judėjimo trajektorijų brėžiamų plotų, siekiant išvengti svirčių susidūrimo su kliūtimis. Kadangi šios situacijos būna retai, tai šiame darbe šie skaičiavimai nėra atliekami.

### 3.1. Kliūčių pozicijos įvertinimas

Šiame darbe tikrinama ar kliūtis yra svirties  $AK$  judėjimo trajektorijoje, t.y. ar kliūtis yra keturkampyje  $AKA_1K_1$ . Taigi įvedamas apribojamas - kliūtis negali būti  $AB$  svirties judėjimo trajektorijoje. Kadangi skaičiavimai tiek  $AK$ , tiek  $AB$  svirtyse šio skyriaus skaičiavimai ir jų taikymai yra analogiški, todėl šiame darbe kliūčių pozicijos vertinimas ir išvengimas analizuojamas tik  $AK$  svirčiai.



21 pav. Manipulatoriaus svirties  $AK$  judėjimo trajektorija

$AKA_1K_1$  kampų koordinatės ir paaiškinimas iš kur randamos (Lentelė 4):

$A$  ir  $A_1$  koordinatėms skaičiavimas išvestas skyriuje „Tiesinės kinematikos sprendimas“

Lentelė 4.  $AKA_1K_1$  kampų koordinatės

Svirties pabaigos ar jungimosi taškas	Koordinatės	Koordinatėms radimas
$A$	$(x_A y_A)$	$x_A = L \cos \gamma + L \cos ABJ$ $y_A = L \sin \gamma + L \sin ABJ$
$A_1$	$(x_{A1} y_{A1})$	$x_{A1} = L \cos \gamma + L \cos A_1B_1J_1$ $y_{A1} = L \sin \gamma + L \sin A_1B_1J_1$
$K$	$(x_1 y_1)$	Manipulatoriaus griebtuvo koordinatės
$K_1$	$(x_2 y_2)$	Užduoties taško koordinatės

Žinant šias koordinatas atitinkamai yra sudaromos tiesinės funkcijos (žiūrėti 21 pav.):

$f_1(x)$  apibrėžiama pagal tiesės  $AK$  taškus;

$f_2(x) - A_1K_1$ ;

$f_3(x) - AA_1$ ;

$f_4(x) - KK_1$ ;

Pagal turimos tiesės dviejų taškų koordinatas yra išvedamos tiesinių funkcijų formulės:

$$f_1(x) = y_3 + \frac{y_1 - y_3}{x_1 - x_3}(x - x_3) \quad (24)$$

$$f_2(x) = y_4 + \frac{y_2 - y_4}{x_2 - x_4}(x - x_4) \quad (25)$$

$$f_3(x) = y_3 + \frac{y_3 - y_4}{x_3 - x_4}(x - x_3) \quad (26)$$

$$f_4(x) = y_1 + \frac{y_1 - y_2}{x_1 - x_2}(x - x_1) \quad (27)$$

Sekančiame žingsnyje tikrinama ar kliūtis (taškas  $H$ ) yra tarp funkcijų  $f_1(x)$  ir  $f_2(x)$  bei  $f_3(x)$  ir  $f_4(x)$ . Jei abu tikrinimai teigiami, tai galima teigti, jog kliūtis yra manipulatoriaus svirties  $AK$  judėjimo trajektorijoje. Jei bent vienas iš tikrinimų yra neigiamas – kliūtis nepatenka į svirties  $AK$  judėjimo trajektoriją.

Pirmiausia tikrinama ar kliūtis yra tarp funkcijų  $f_1(x)$  ir  $f_2(x)$  (žiūrėti 21 pav.) Tam tikslui reikia rasti kokioje koordinačių ašyje, abscisėje ar ordinatėje, kliūtis padėtų riboja šios funkcijos. Tikrinama koku kampu, ar mažesniu už  $45^\circ$ , pasirinktinai bet kuri iš šių funkcijų (pasirinktinai šiuo atveju  $f_1(x)$ ) pasvirusi abscisės atžvilgiu. Jei kampas mažesnis už  $45^\circ$ , tai tiesinės funkcijos koeficientas  $k$  yra mažesnis už vienetą:

$$\frac{y_1 - y_3}{x_1 - x_3} < 1 \quad (28)$$

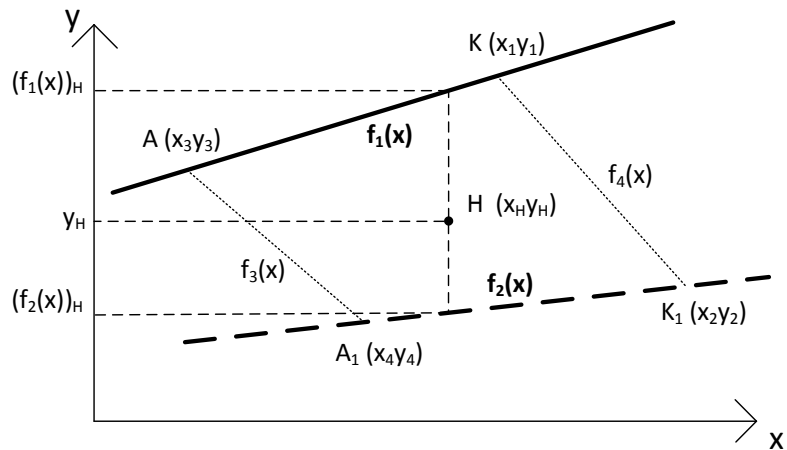
Jei nelygė teisinga, tai funkcija su abscese sudaro mažesnę nei  $45^\circ$  kampą, tai funkcija  $f_1(x)$  riboja taško  $H$  reikšmę ordinačių ašyje.

Taigi, jei nelygė teisinga, tai į funkcijas  $f_1(x)$  ir  $f_2(x)$  įsistatome  $H$  taško  $x$  koordinatę:

$$(f_1(x))_H = y_3 + \frac{y_1 - y_3}{x_1 - x_3}(x_H - x_3) \quad (29)$$

$$(f_2(x))_H = y_4 + \frac{y_2 - y_4}{x_2 - x_4}(x_H - x_4) \quad (30)$$

Iš šių lygčių randamos ordinačių reikšmės (22 pav.)



22 pav. Kliūtės pozicijos tikrinimas funkcijų  $f_1(x)$  ir  $f_2(x)$  atžvilgiu

Sekančioje nelygybėje tikrinama ar kliūtis (taškas  $H$ ) yra tarp šių funkcijų, t.y. ar taško  $H$  ordinatės reikšmė yra daugiau už  $f_2(x)_H$ , bet mažiau už  $f_1(x)_H$ :

$$(f_2(x))_H < y_H < (f_1(x))_H \quad (31)$$

Jei nelygybė neteisinga, tai kliūtis taškas nėra tarp funkcijų  $f_1(x)$  ir  $f_2(x)$  ir galima teigti, jog svirties  $AK$  judėjimo trajektorijoje kliūtis nėra. Jei nelygybė teisinga – tikriname ar taškas  $H$  yra tarp funkcijų  $f_3(x)$  ir  $f_4(x)$ .

Jei funkcijos  $f_1(x)$  ir  $f_2(x)$  kliūtis tašką riboja ordinačių ašyje, tai funkcijos  $f_3(x)$  ir  $f_4(x)$  tašką  $H$  ribos abscisės ašyje ir atvirkščiai - jei funkcijos  $f_1(x)$  ir  $f_2(x)$  kliūtis tašką riboja abscisės ašyje, tai funkcijos  $f_3(x)$  ir  $f_4(x)$  ribos ordinatės ašyje. Norint patikrinti ar kliūtis yra tarp funkcijų  $f_3(x)$  ir  $f_4(x)$  reikia į jas įsistatyti kliūtis taško  $y$  koordinatę.

Įsistatome  $y_H$  koordinatę į  $f_3(x)$  funkciją:

$$y_H = y_3 + \frac{y_3 - y_4}{x_3 - x_4} (x - x_3) \quad (32)$$

Iš šios formulės išsireiškiame  $x$  ir pažymime jį  $x_{H3}$ :

$$x_{H3} = \frac{(y_H - y_3)(x_3 - x_4)}{y_3 - y_4} + x_3 \quad (33)$$

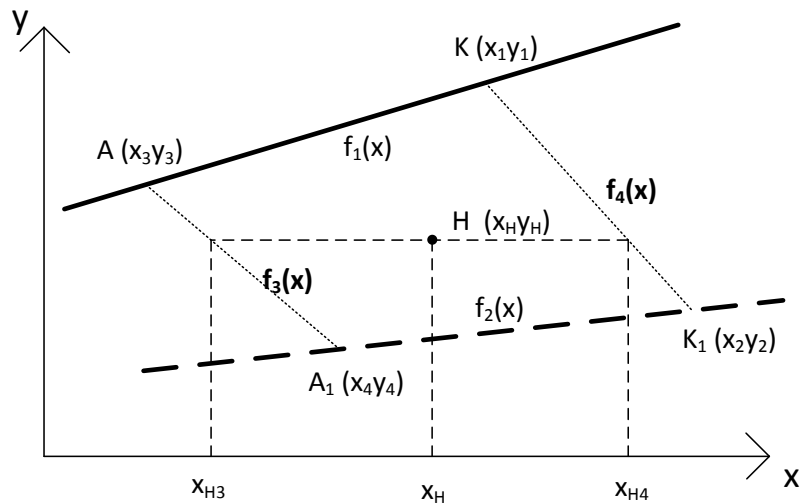
Įsistatome  $y_H$  koordinatę į  $f_4(x)$  funkciją:

$$y_H = y_1 + \frac{y_1 - y_2}{x_1 - x_2} (x - x_1) \quad (34)$$

Iš šios formulės išsireiškiame  $x$  ir pažymime jį  $x_{H4}$ :

$$x_{H4} = \frac{(y_H - y_1)(x_1 - x_2)}{y_1 - y_2} + x_1 \quad (35)$$

Iš šių lygčių  $x_{H3}$  ir  $x_{H4}$  randamos abscisių reikšmės (žiūrėti 23 pav.)



23 pav. Kliūtis pozicijos tikrinimas funkcijų  $f_3(x)$  ir  $f_4(x)$  atžvilgiu

Turint  $x_{H3}$  ir  $x_{H4}$  reikšmes nelygybėje (25) tikrinama ar kliūtis taškas yra tarp funkcijų  $f_3(x)$  ir  $f_4(x)$  – t.y. ar taško  $H$  koordinatės  $x$  vertė yra daugiau už  $x_{H3}$ , bet mažiau už  $x_{H4}$ .

$$x_{H3} < x_H < x_{H4} \quad (36)$$

Jei nelygybė neteisinga – kliūtis taškas  $H$  nėra tarp funkcijų  $f_3(x)$  ir  $f_4(x)$  ir galima teigti jog kliūtis nėra svirties  $AK$  judėjimo trajektorijoje. Jei nelygybė teisinga – galime daryti išvadą, jog kliūtis yra svirties  $AK$  judėjimo trajektorijoje ir reikalinga taikyti kliūčių išvengimo metodą.

Taigi, išanalizavus įvairias kliūtis padėties situacijas manipulatoriaus svirties  $AK$  atžvilgiu, jos yra pateikiamos Lentelė 5:

Simbolis – žalia „varnelė“ reiškia, kad kliūtis tenkina užduotas sąlygas;

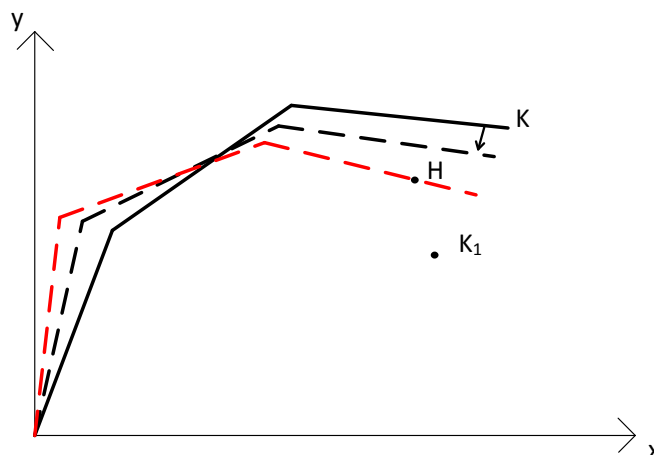
Simbolis – raudonas „x“ reiškia, kad kliūtis nėra tarp užduotų sąlygų;

Lentelė 5. Kliūtis padėties variantai svirties AK atžvilgiu

Kliūtis pozicija	Funkcijos $f_i(x)$ koeficientas - $k < 1$			
Tarp $f_1(x)$ ir $f_2(x)$	$\begin{cases} (f_2(x))_H < y_H \\ (f_1(x))_H > y_H \end{cases}$		$\emptyset \begin{cases} (f_2(x))_H < y_H \\ (f_1(x))_H > y_H \end{cases}$	
Tarp $f_3(x)$ ir $f_4(x)$	$\begin{cases} x_{H3} < x_H \\ x_{H4} > x_H \end{cases}$	$\emptyset \begin{cases} x_{H3} < x_H \\ x_{H4} > x_H \end{cases}$	$\begin{cases} x_{H3} < x_H \\ x_{H4} > x_H \end{cases}$	$\emptyset \begin{cases} x_{H3} < x_H \\ x_{H4} > x_H \end{cases}$
Svirties AK trajektorijoje				
Kliūtis pozicija	Funkcijos $f_i(x)$ koeficientas - $k > 1$			
Tarp $f_1(x)$ ir $f_2(x)$	$\begin{cases} x_{H1} < x_H \\ x_{H2} > x_H \end{cases}$		$\emptyset \begin{cases} x_{H1} < x_H \\ x_{H2} > x_H \end{cases}$	
Tarp $f_3(x)$ ir $f_4(x)$	$\begin{cases} (f_3(x))_H < y_H \\ (f_4(x))_H > y_H \end{cases}$	$\emptyset \begin{cases} (f_3(x))_H < y_H \\ (f_4(x))_H > y_H \end{cases}$	$\begin{cases} (f_3(x))_H < y_H \\ (f_4(x))_H > y_H \end{cases}$	$\emptyset \begin{cases} (f_3(x))_H < y_H \\ (f_4(x))_H > y_H \end{cases}$
Svirties AK trajektorijoje				

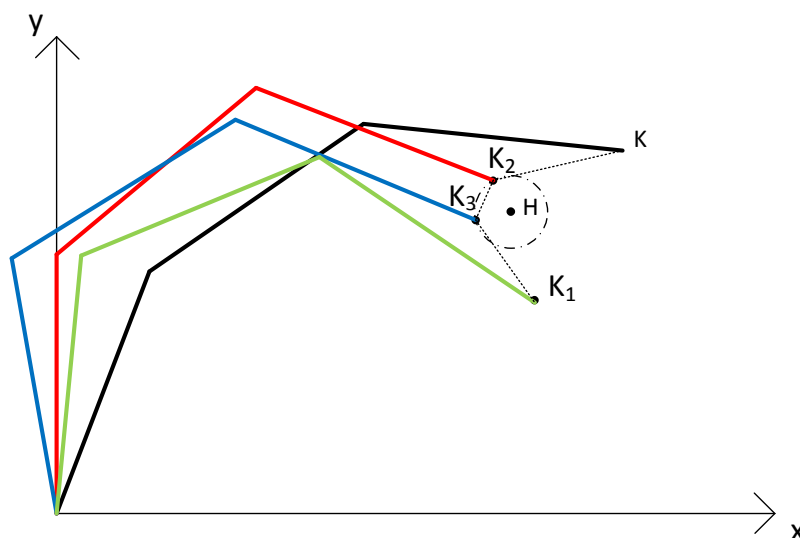
### 3.2. Kliūčių išvengimo metodas

Išanalizavus kliūtis poziciją ir nustatius, jog kliūtis yra manipulatoriaus svirties judėjimo trajektorijoje (24 pav.), toliau naudojamas kliūtis išvengimo metodas. Manipulatoriaus griebtuvo taškas  $K$  kliūtį visada bando aplenkti iš savo įtvirtinimo ašies pusės, nes iš kitos pusės tai padaryti yra neįmanoma – pagal nustatytas sąlygas manipulatoriaus svirčių tarpusavės padėtys kinta nuo  $0^0$  iki  $180^0$ .



24 pav. Kliūtis manipulatoriaus svirties trajektorijoje

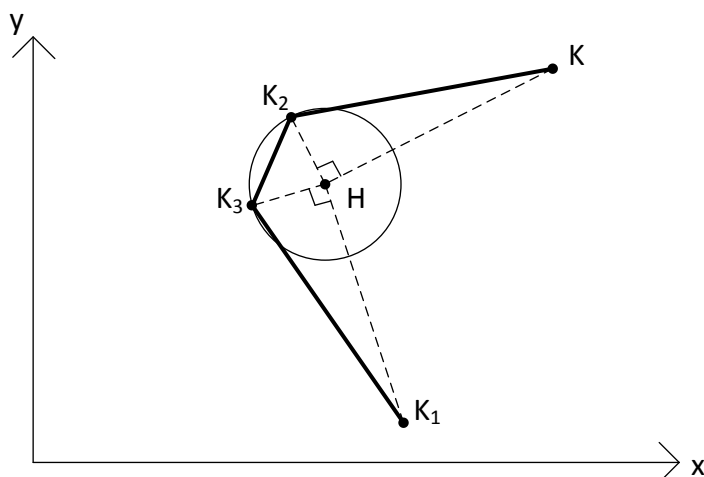
Naudojant kliūtis išvengimo metodą iš pradžių apie kliūtį yra apibrėžiamas apskritimas, kurio spindulys  $L/7$  ( $L$  – tai svirties ilgis). Metodo esmė – manipulatoriaus griebtuvui yra sukuriami 2 tarpiniai taškai apibrėžto apskritimo lanke, per kuriuos griebtuvui judant nuo pradinės būsenos  $K$  iki užduoties įvykdymo padėties  $K_1$  yra išvengiamas svirties susidūrimas su kliūtimi ().



25 pav. Kliūtis išvengimo metodas

Paveiksle juodomis linijomis pavaizduotos svirtys pradinėje padėtyje. Raudonomis linijomis – svirčių padėtys griebtuvui pasiekus 1-ąjį tarpinį tašką. Mėlynomis – 2-ąjį tarpinį tašką ir žaliomis – svirčių padėtys pasiekus užduoties tašką.

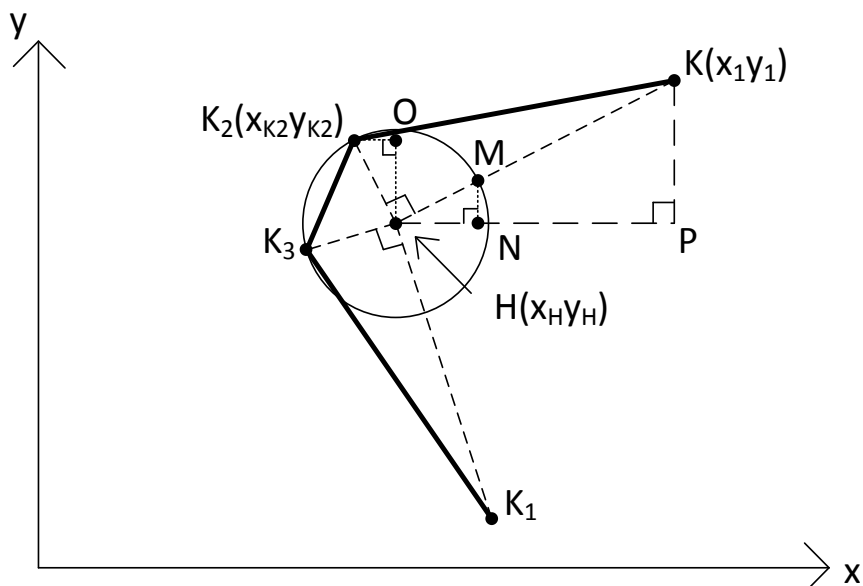
Tarpinis taškas  $K_2$  (25 pav.) randamas nubrėžus tiesę tarp griebtuvo pradinę padėties  $K$  ir kliūtis  $H$  (26 pav.). Šiai tiesei nubrėžiamas statmuo nuo taško  $H$  iki apskritimo kreivės. Statmens ir apskritimo kreivės jungimosi taškas  $K_2$  – tai pirmasis griebtuvo tarpinis taškas. Antrasis griebtuvo tarpinis taškas randamas analogiškai: nubrėžus tiesę tarp griebtuvo užduoties taško  $K_1$  ir kliūtis  $H$  ir šiai atkarpai išvedus statmenį nuo taško  $H$  iki apskritimo lanko. Statmens ir lanko sankirtos taškas – 2-asis tarpinis taškas.



26 pav. Griebtuvo trajektorijos tarpiniai taškai

Taigi, siekiant išvengti kliūties ir manipulatoriaus svirties susidūrimo, griebtuvo trajektorija brėžiama per tarpinius taškus, tam tikslui yra randamos taškų  $K_2$  ir  $K_3$  koordinatės.

Siekiant surasti  $K_2$  taško koordinatės, iš pradžių yra randamos taško  $M$  koordinatės nusakančių tiesių  $MN$  ir  $HN$  skaitinės reikšmės (27 pav.):



27 pav. Tarpinio taško  $K_2$  koordinačių radimas

Iš paveikslo matome, jog taško  $M$  koordinatės gaunamos prie  $H$  koordinačių pridėjus atitinkamai atkarpų  $MN$  ir  $HN$  skaitines vertes (prie  $H$  taško abscisės pridėjus  $HN$ , prie ordinatės  $MN$ ). Tiesės  $MN$  ir  $HN$  yra skaičiuojamos lyginant panašius trikampius  $MHN$  ir  $KHP$ . Ieškomas koeficientas  $k$  - santykis tarp  $HM$  ir  $KH$ :

$$KH = \sqrt{(x_1 - x_H)^2 + (y_1 - y_H)^2} \quad (37)$$

$$HM = L/7 \quad (38)$$

$$k = \frac{HM}{KH} = \frac{L}{7\sqrt{(x_1 - x_H)^2 + (y_1 - y_H)^2}} \quad (39)$$

Pagal koeficientą  $k$  surandamos tiesės  $MN$  (lyginant su  $KP$ ) ir  $HN$  (lyginant  $HP$ ):

$$MN = k(x_1 - x_H) = \frac{L(x_1 - x_H)}{7\sqrt{(x_1 - x_H)^2 + (y_1 - y_H)^2}} \quad (40)$$

$$HN = k(y_1 - y_H) = \frac{L(y_1 - y_H)}{7\sqrt{(x_1 - x_H)^2 + (y_1 - y_H)^2}} \quad (41)$$

Trikampiai  $MHN$  ir  $K_2HO$  vienodi:  $K_2O = MN$ ,  $OH = HN$ . Taigi taško  $K_2$  koordinatės randamos prie taško  $H$  koordinačių atitinkamai pridėjus tieses  $K_2O$  ir  $OH$  (iš taško  $H$  abscisės atimama tiesės  $K_2O$  skaitinė vertė, prie taško  $H$  ordinatės pridedama tiesės  $OH$  skaitinė vertė):



Trikampiai  $CHF$  ir  $K_3HE$  vienodi:  $K_3E = HF$ ,  $EH = CF$ . Taigi taško  $K_3$  koordinatės randamos prie taško  $H$  koordinatėms atitinkamai pridėjus tieses  $K_3E$  ir  $EH$  (iš taško  $H$  ordinatės pridėjama tiesės  $K_3E$  skaitinė vertė, prie taško  $H$  abscisės atimama tiesės  $EH$  skaitinė vertė):

$$x_{K_3} = x_H - EH = x_H - \frac{L(x_2 - x_H)}{7\sqrt{(x_2 - x_H)^2 + (y_2 - y_H)^2}} \quad (49)$$

$$y_{K_3} = y_H + K_3E = y_H + \frac{L(y_2 - y_H)}{7\sqrt{(x_2 - x_H)^2 + (y_2 - y_H)^2}} \quad (50)$$

Taigi, naudojant kliūčių išvengimo metodą išvedamos tarpinių taškų skaičiavimo formulės (Lentelė 6); per šiuos taškus nubrėžiama manipulatoriaus griebtuvo judėjimo trajektorija. Tokiu būdu griebtuvas aplenkia kliūtį jos nekliudant nei griebtuvui nei svirčiai ir pasiekia užduotą tikslą.

Lentelė 6. Griebtuvo tarpinių taškų koordinatės

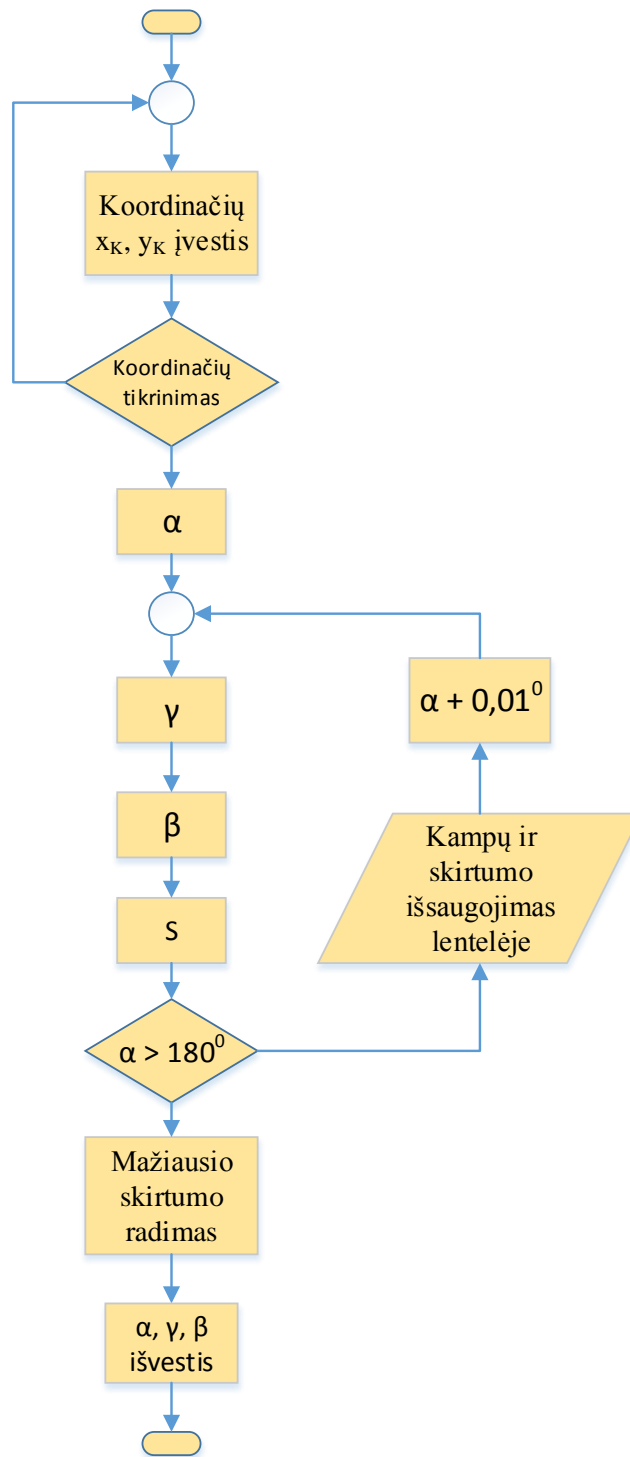
Tarpinis taškas	Abscisė	Ordinatė
<b>K<sub>2</sub></b>	$x_{K_2} = x_H - \frac{L(x_1 - x_H)}{7\sqrt{(x_1 - x_H)^2 + (y_1 - y_H)^2}}$	$y_{K_2} = y_H + \frac{L(y_1 - y_H)}{7\sqrt{(x_1 - x_H)^2 + (y_1 - y_H)^2}}$
<b>K<sub>3</sub></b>	$x_{K_3} = x_H - \frac{L(x_2 - x_H)}{7\sqrt{(x_2 - x_H)^2 + (y_2 - y_H)^2}}$	$y_{K_3} = y_H + \frac{L(y_2 - y_H)}{7\sqrt{(x_2 - x_H)^2 + (y_2 - y_H)^2}}$

#### 4. Manipulatoriaus valdymo sistemos algoritmas

##### 4.1. Atvirkštinės kinematikos skaičiavimų algoritmas

Atvirkštinės kinematikos skaičiavimų algoritmas (29 pav.) naudojamas rasti manipulatoriaus svirčių tarpusavių padėčių kampų vertes pagal manipulatoriaus griebtuvo padėtį koordinatėms ašyje.

Kadangi manipulatorius turi 3 laisvės laipsnių koordinatėms plokštumoje - svirčių tarpusavės padėties kampų variantų gali būti begalė [18]. Algoritmo pagalba yra apskaičiuojami visi įmanomi variantai pagal užduotą kampo  $\alpha$  kitimo vertę. Kuo mažesnis kampo  $\alpha$  kitimas, tuo daugiau variantų algoritmas pateikia. Šiame darbe tiriamas manipulatorius, kurio kampo  $\alpha$  kitimas užduotas  $0,01^\circ$ . Gautus svirčių padėčių variantus algoritmas lygina su esamomis manipulatoriaus svirčių padėtimis prieš atliekant užduotį ir išrenka variantą, kuris yra energetiškai efektyviausias užduočiai atlikti, t.y. svirčių tarpusavės padėties kampų pokyčių suma yra mažiausia.



29 pav. Atvirkštinės kinematikos skaičiavimų algoritmas

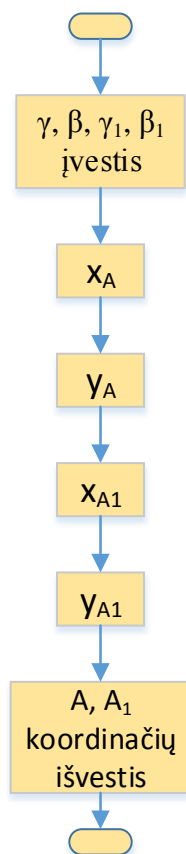
Gavus užduoties koordinates algoritmas tikrina ar užduotis yra manipulatoriaus veikimo zonoje (29 pav.), t.y. atstumas  $KO$  nuo manipulatoriaus įtvirtinimo taško iki užduoties (17 pav.):  $3L > KO > L$ . Jei sąlyga netenkinama algoritmas laukia kitų koordinačių įvesties, jei sąlyga teisinga, skaičiuojami kampai:  $\alpha$ ,  $\gamma$ ,  $\beta$  ir kampų pokyčių suma  $S$ . Atlikus skaičiavimus tikrinama ar  $\alpha > 180^\circ$ , nes tai maksimali kampo  $\alpha$  vertė kuria gali manipulatorius funkcionuoti. Kol sąlyga nėra tenkinama, algoritmas išsaugo skaičiavimų rezultatus lentelėje ir kampą  $\alpha$  padidinus  $0,01^\circ$  skaičiavimų ciklas kartojamas. Kai kampo  $\alpha$  vertė tampa didesnė už  $180^\circ$  algoritmas randa išsaugotų skaičiavimų

mažiausią kampų pokyčių sumos vertę ir kampus pagal kuriuos ši vertė paskaičiuota. Kampai yra išvedami tolimesniems skaičiavimams.

#### 4.2. Tiesinės kinematikos skaičiavimų algoritmas

Tiesinės kinematikos skaičiavimų algoritmas (30 pav.29 pav.) naudojamas rasti manipulatoriaus antrosios ir trečiosios (skaičiuojant nuo pagrindo) svirties jungimosi ašies padėtį koordinačių ašyje. Manipuliatoriui esant pradinėje būsenoje ašis žymima  $A$  tašku, o manipuliatoriui esant užduoties įvykdymo pozicijoje  $A_1$  tašku (19 pav.). Šių taškų koordinačių skaičiavimas reikalingas tolimesniems kliūtis pozicijos įvertinimo skaičiavimams.

Algoritmui nuskaičius kampų  $\gamma, \beta, \gamma_1, \beta_1$  vertes yra skaičiuojamos  $x_A, y_A$  ir  $x_{A1}, y_{A1}$  koordinačių vertės. Radus šias koordinates, jos galutiniame variante yra išvedamos kaip taškų  $A$  ir  $A_1$  koordinatės, skirtos tolimesniems skaičiavimams.

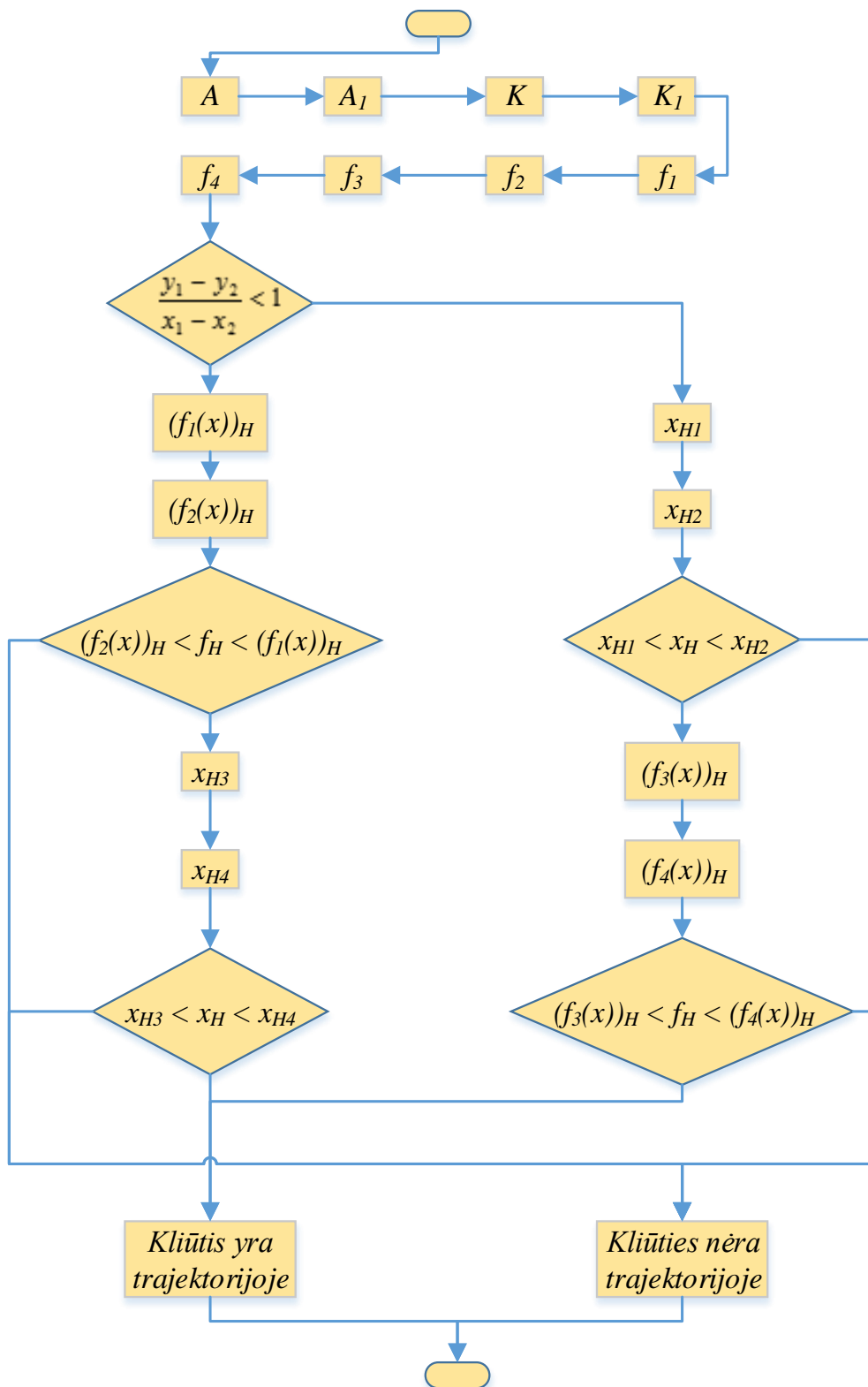


30 pav. Tiesinės kinematikos skaičiavimų algoritmas

Algoritmo kampų  $\gamma, \beta, \gamma_1, \beta_1$  įvesties dalyje, nėra šių kampų korektiškumo tikrinimo, nes jie yra išskaičiuoti iš atvirkštinės kinematikos skaičiavimo algoritmo (29 pav.) ir galima teigti, jog kampai yra korektiški.

### 4.3. Kliūčių pozicijos įvertinimo algoritmas

Kliūčių pozicijos įvertinimo algoritmas (31 pav.) pagal kliūtis koordinatės įvertina ar kliūtis patenka į manipulatoriaus judėjimo trajektoriją. Jei patenka – naudojamas kliūtis išvengimo algoritmas, jei nepatenka – manipulatoriaus svirčių kampai pasukami atitinkamai  $\alpha$ ,  $\gamma$ ,  $\beta$  kampais, apskaičiuotais atvirkštinės kinematikos skaičiavimo algoritmo.



31 pav. Kliūtis pozicijos įvertinimo algoritmas

Algoritmas nuskaito  $A$ ,  $A_l$  koordinatinių vertes iš tiesinės kinematikos algoritmo.  $K$ ,  $K_l$  koordinatės yra žinomos iš sąlygos. Žinant koordinatinių vertes atitinkamai yra sudaromos manipulatoriaus svirčių ir menamų svirčių jungimosi tiesių funkcijos  $f_1(x)$ ,  $f_2(x)$ ,  $f_3(x)$ ,  $f_4(x)$ .

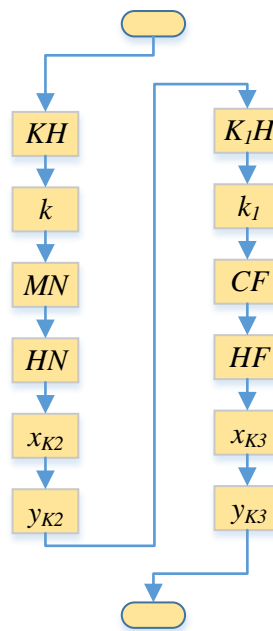
Sekančiame žingsnyje tikrinama ar kliūtis yra tarp funkcijų  $f_1(x)$  ir  $f_2(x)$ . Tam tikslui reikia rasti kokioje koordinatinių ašyje, abscisėje ar ordinatėje, kliūtis padėtų riboja šios funkcijos. Tikrinama kokių kampų, ar mažesniu už  $45^\circ$ , pasirinktinai bet kuri iš šių funkcijų (pasirinktinai šiuo atveju  $f_1(x)$ ) pasvirusi abscisės atžvilgiu. Jei kampas mažesnis už  $45^\circ$ , tai tiesinės funkcijos koeficientas  $k$  yra mažesnis už vienetą.

Jei nelygė teisinga, tai funkcija su abscise sudaro mažesnę nei  $45^\circ$  kampą - funkcija  $f_1(x)$  riboja taško  $H$  reikšmę ordinačių ašyje ir toliau yra skaičiuojamos funkcijų  $f_1(x)$ ,  $f_2(x)$  ordinačių vertės  $x_H$  abscisės atžvilgiu. Funkcijų  $f_1(x)$ ,  $f_2(x)$  ir kliūtis ordinatės yra lyginamos, t.y. ar kliūtis ordinatės vertė yra tarp funkcijų ordinačių. Jei taip – toliau analogiškai tikrinama ar kliūtis abscisė yra tarp funkcijų  $f_3(x)$ ,  $f_4(x)$  abscisių. Jei sąlyga tenkinama, tai išvedamas teiginys, kad kliūtis yra manipulatoriaus svirčių judėjimo trajektorijoje ir toliau naudojamas kliūtis išvengimo algoritmas. Jei bent viena iš šių dviejų koordinatinių tikrinimų sąlyga neteisinga, tai gaunama išvada, kad kliūtis nėra manipulatoriaus judėjimo trajektorijoje ir manipulatoriaus svirtys pasukamos atitinkamai atvirkštinės kinematikos algoritme apskaičiuotais kampais  $\alpha$ ,  $\gamma$ ,  $\beta$ .

Jei nelygė neteisinga, tai funkcija su abscise sudaro didesnę nei  $45^\circ$  kampą - funkcija  $f_1(x)$  riboja taško  $H$  reikšmę abscisės ašyje ir toliau yra skaičiuojamos funkcijų  $f_1(x)$ ,  $f_2(x)$  abscisių vertės  $x_H$  ordinatės atžvilgiu. Funkcijų  $f_1(x)$ ,  $f_2(x)$  ir kliūtis abscisės yra lyginamos, t.y. ar kliūtis abscisės vertė yra tarp funkcijų abscisių. Jei taip – toliau analogiškai tikrinama ar kliūtis ordinatė yra tarp funkcijų  $f_3(x)$ ,  $f_4(x)$  ordinačių. Jei sąlyga tenkinama, tai išvedamas teiginys, kad kliūtis yra manipulatoriaus svirčių judėjimo trajektorijoje ir toliau naudojamas kliūtis išvengimo algoritmas. Jei bent viena iš šių dviejų koordinatinių tikrinimų sąlyga neteisinga, tai gaunama išvada, kad kliūtis nėra manipulatoriaus judėjimo trajektorijoje ir manipulatoriaus svirtys pasukamos atitinkamai atvirkštinės kinematikos algoritme apskaičiuotais kampais  $\alpha$ ,  $\gamma$ ,  $\beta$ .

#### 4.4. Kliūčių išvengimo algoritmas

Išanalizavus kliūtis poziciją manipulatoriaus atžvilgiu ir nustatius, jog kliūtis yra tarp manipulatoriaus svirčių judėjimo trajektorijų taikomas kliūtis išvengimo algoritmas (32 pav.). Kliūtis išvengimo algoritmo esmė – tai sukuriama 2 tarpiniai taškai manipulatoriaus judėjimo trajektorijoje, nutolę nuo kliūtis  $L/7$  spinduliu ( $L$  – svirties ilgis).



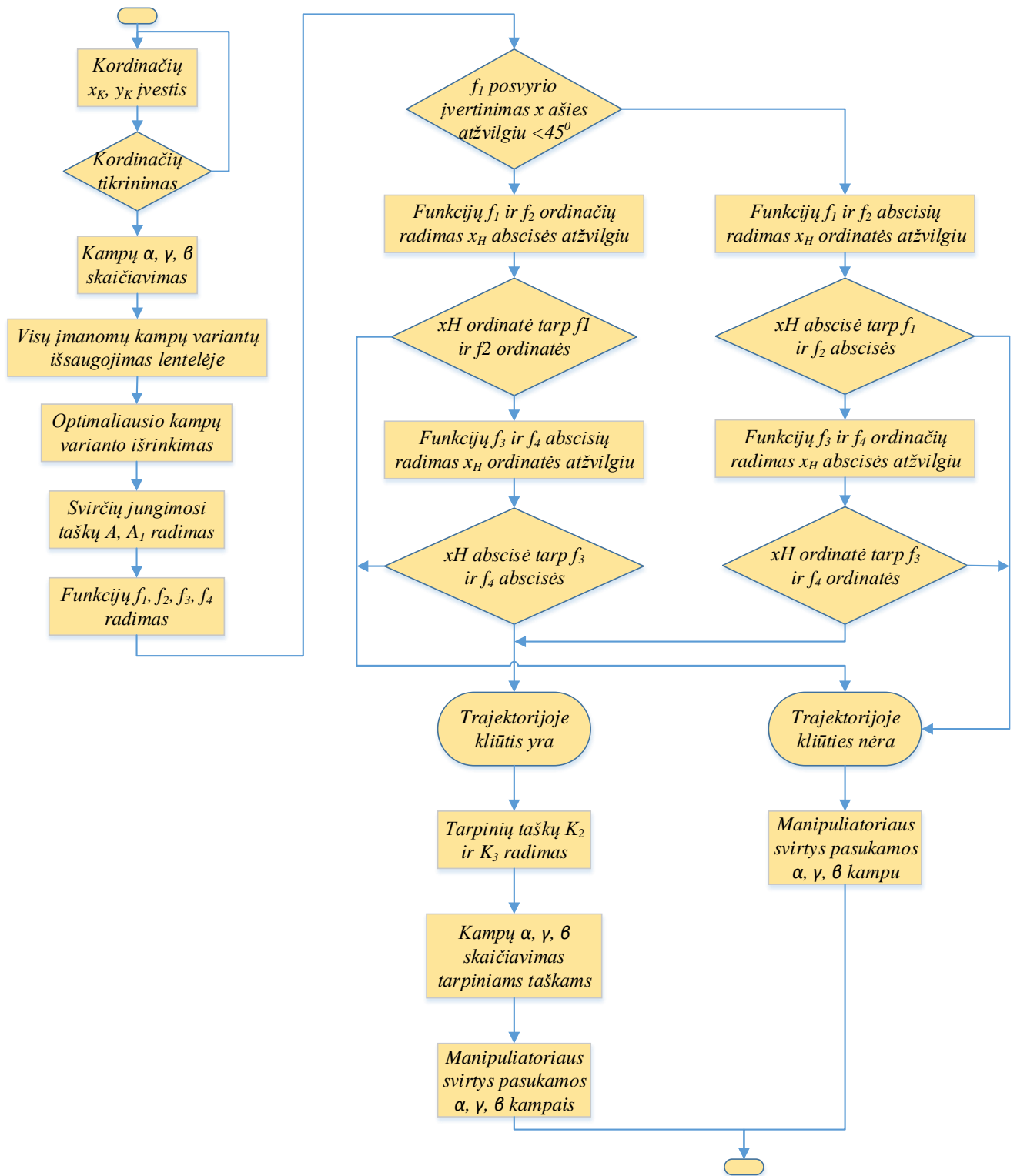
32 pav. Kliūties išvengimo algoritmas

Algoritmo pirmame žingsnyje yra apskaičiuojamas atstumas nuo svirties pabaigos taško iki kliūtis –  $KH$ . Toliau skaičiuojamas koeficientas  $k$  – santykis tarp  $KH$  ir  $L/7$ . Sekančiame žingsnyje apskaičiuojamos tiesės  $MN$  ir  $HN$ , kurios nurodo atstumą  $x$  ir  $y$  ašimi nuo tarpinio taško. Žinant šiuos atstumus yra paskaičiuojamos pirmojo tarpinio taško koordinatės  $x_{K2}$  ir  $y_{K2}$ . Plačiau apie kliūtis išvengimo metodą – „Kliūtis išvengimo metodas“ skyriuje.

Atrojo tarpinio taško  $K_3$  koordinatės  $x_{K3}$  ir  $y_{K3}$  randamos analogiškai.

#### 4.5. Manipulatoriaus valdymo sistemos algoritmas

Manipulatoriaus valdymo sistemos algoritmas (33 pav.) aprašo bendra visos valdymo sistemos abstraktų veikimo principą. Sukūrus šį algoritmą aiškiai matosi visi žingsniai kaip skaičiuojami manipulatoriaus svirčių tarpusavės padėties kampai bei atliekamas kliūtis pozicijos manipulatoriaus svirčių atžvilgiu įvertinimas ir jei kliūtis padėtis nustatoma manipulatoriaus svirčių judėjimo trajektorijoje – atliekamas kliūtis išvengimas, naudojant tarpinius taškus, per kuriuos judėdamas manipulatoriaus griebtuvas išvengia kliūtis susidūrimo ir sėkmingai pasiekia užduotį.



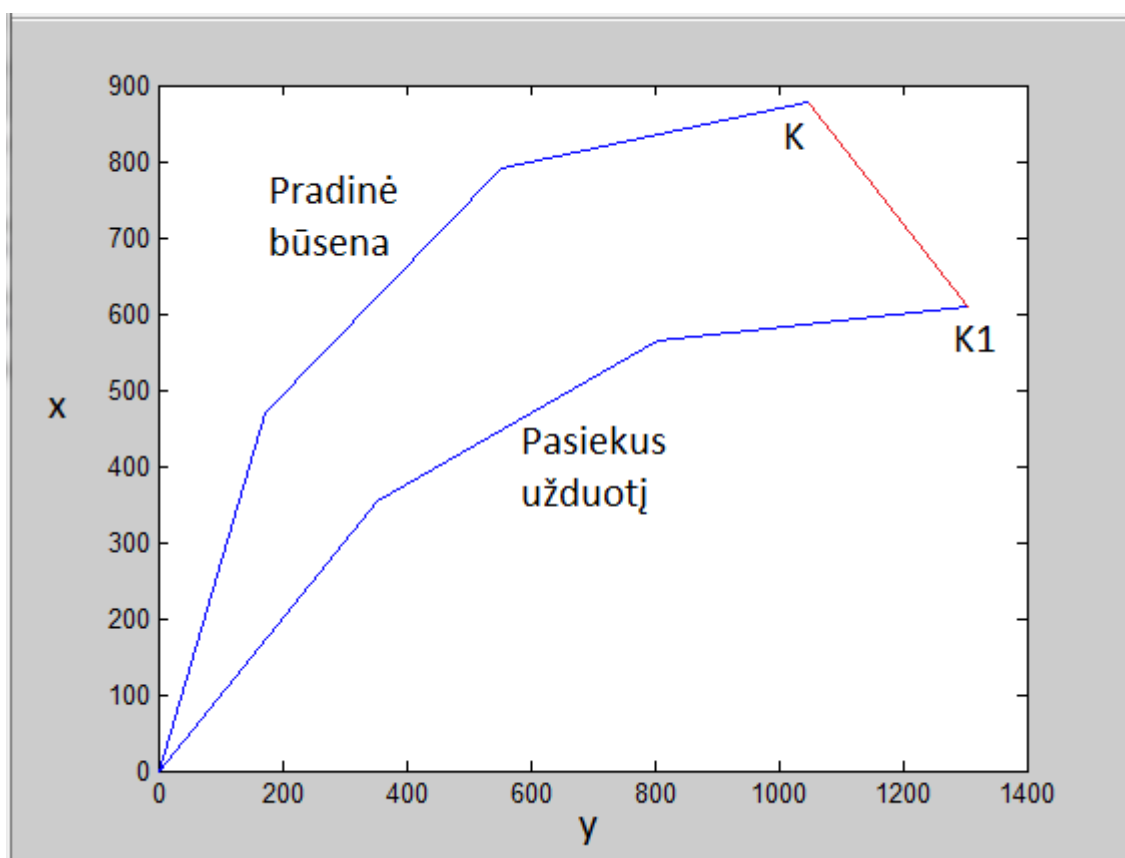
33 pav. Manipulatoriaus valdymo sistemos algoritmas

Remiantis šiuo algoritmu buvo sukurtas manipulatoriaus išmaniosios valdymo sistemos programinis kodas.

## IV. EKSPERIMENTINĖ DALIS

### 1. Manipulatoriaus eksperimentiniai tyrimai

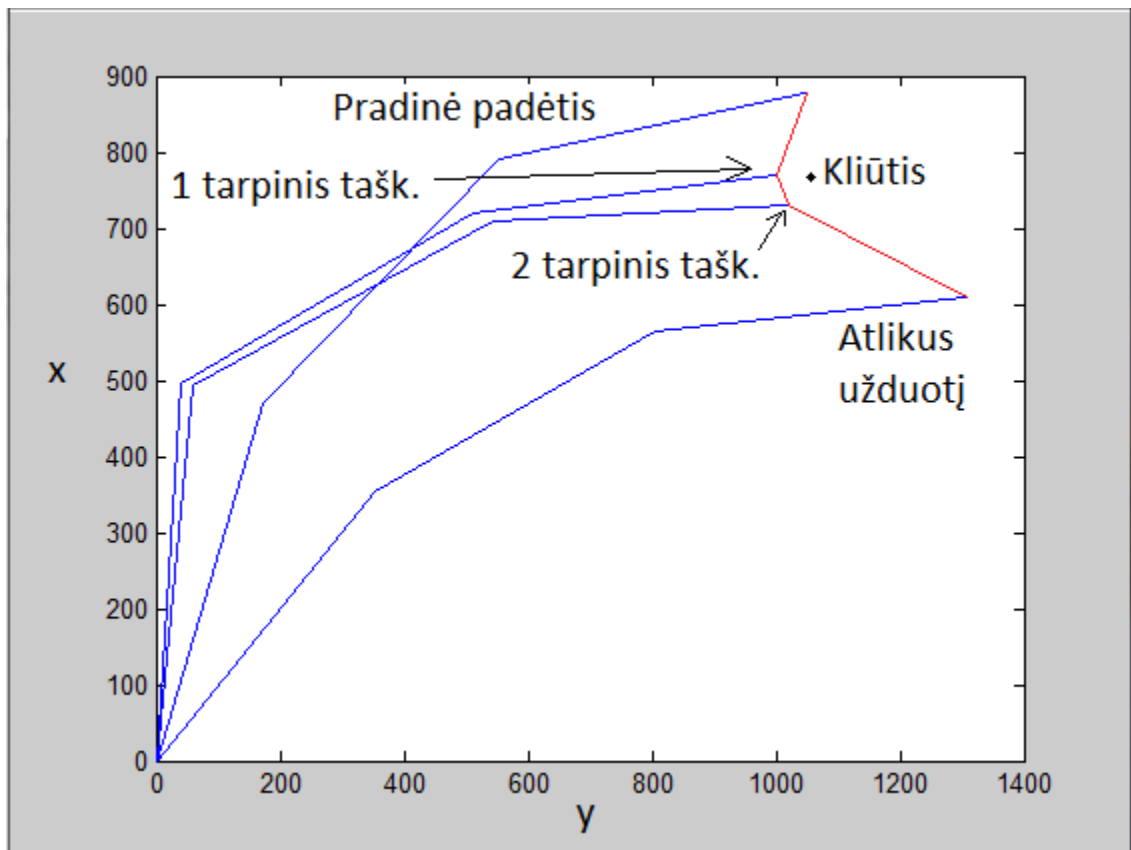
Eksperimentas buvo atliekamas virtualiai sumodeliavus manipulatoriaus išmaniosios valdymo sistemos užduoties atlikimą. C++ programoje atlikus kinematinius skaičiavimus bei pritaikius kliūčių įvertinimo ir išvengimo metodus gautos manipulatoriaus svirčių jungimo taškų koordinatės. Šias koordinates įvedus į „Matlab“ programą naudojant „plot“ funkciją buvo atvaizduotos manipulatoriaus svirtys.



34 pav. Manipulatoriaus užduoties atlikimas „Matlab“

Manipulatoriaus svirtys (34 pav.) pavaizduotos prieš atliekant užduotį, t.y. pradinėje būsenoje (viršutinė padėtis) ir atlikus užduotį (apatinė padėtis). Kai kliūtis nėra manipulatoriaus svirčių judėjimo trajektorijoje, tai panaudojus kinematinius skaičiavimus yra parenkamas mažiausiai energetinių resursų naudojantis svirčių tarpusavės padėties kampų variantas, t.y. parenkama mažiausia kampų pokyčių suma.

36 paveiksle analogiškai „Matlab“ programoje atvaizduotos svirčių padėtys. Svirčių jungimo taškų koordinatės randamos C++ programoje integravus kliūčių įvertinimo ir išvengimo metodus.



35 pav. Manipulioriaus užduoties atlikimas esant kliūčiai „Matlab“

Atsiradus kliūčiai manipulatoriaus svirčių judėjimo trajektorijoje, išmanioji sistema įvertino, kad reikia keisti manipulatoriaus judėjimo trajektoriją. Vadovaujantis kliūčių išvengimo metodu sistema atliko tarpinių griebtuvo judėjimo taškų koordinatinių skaičiavimus ir sėkmingai parinko naują trajektoriją kliūties apėjimui. Trajektorijos daliai tarp kiekvieno taško yra taikomi kinematiniai skaičiavimai, siekiant rasti pačią optimaliausią trajektoriją, naudojančią mažiausiai energetinių resursų.

Atlikus šį eksperimentą pastebėta, jog darbo eigoje aprašyti kinematiniai skaičiavimai bei kliūties analizės ir išvengimo metodai yra korektiški ir gali būti taikomi kuriant fizinius manipulatoriaus išmaniosios valdymo sistemos modelius.

## IŠVADOS

Apžvelgus esamų manipulatoriaus valdymo sistemų, taikomų įvairiose srityse, analogus ir įvertinus jų galimybes, nustatyta jog jau esamos manipulatoriaus išmaniosios valdymo sistemos nėra pakankamai efektyvios ir saugios neplanuotomis aplinkos sąlygomis. Atsižvelgiant į tokių sistemų trūkumus, pasiūlytas sistemos valdymo algoritmas randantis trumpiausią užduoties atlikimo kelią bei reikalui esant padedantis išvengti kliūčių.

Manipulatoriaus virtualaus modelio matematinio aprašo lygčių tyrimas parodė, jog manipulatoriui energetinis efektyvumas ir greitimeika galima padidinti, svirtims judant pagal apskaičiuotąją specialios formos trajektoriją.

Pritaikius kliūtis pozicijos vertinimo ir kliūtis išvengimo būdą paaiškėjo, jog tai leidžia tikslingai atlikti kliūtis padėties analizę ir būtinus veiksmus jai išvengti. Šis būdas gali būti taikomas manipulatoriuose, kuriuose reikalingas aukšto lygio saugumas kliūčių atžvilgiu.

Sukurtosios manipulatoriaus automatinio valdymo sistemos darbingumą ir algoritmo bei programos kodo funkcionalumą patvirtina virtualaus manipulatoriaus eksperimentinio tyrimo rezultatai.

### Darbo tęstinumas

Atlikus manipulatoriaus svirčių kinematinis skaičiavimus bei integravus kliūčių analizės ir išvengimo metodus, eksperimentiškai buvo pagrįstas manipulatoriaus sistemos funkcionalumas. Ateityje būtų galima nagrinėti keturių svirčių manipulatorių, siekiant padidinti sistemos lankstumą bei keletos judančių kliūčių išvengimo galimybes.

### Padėka

Projektui „Traukinys“ – „Intelligent Train Control System“ už praktikos vietą bei galimybę kurti inovatyvias išmaniasias valdymo sistemas.

## LITERATŪRA

1. Šarpnickas, T.; Eglynas, T.; Jankauskas, T.; Vaupšas, J. 2011. Intelektualaus manipulatoriaus modelio programavimas ir optimalios trajektorijos parinkimas, iš 14-osios Lietuvos jaunųjų mokslininkų konferencijos „Mokslas – Lietuvos ateitis“ 2011 metų teminės konferencijos Informatikos sekcijos straipsnių rinkinys. ISBN 978-9955-28-834-3.
2. Šarpnickas, T.; Bulbenkienė, V. 2012. Manipulatoriaus išmaniosios valdymo sistemos kūrimas ir energetinių kaštų minimizavimas, Technologijos mokslo darbai vakarų Lietuvoje VIII: 249. ISSN 1822-4652.
3. Fuhai Zhang, Yili Fu, Shibo Yan, and Shuguo Wang. 2010. Kinematics and simulations of space manipulator in three work modes, Systems and Control in Aeronautics and Astronautics: 572.
4. Hongtai Cheng, Heping Chen, Yong Liu. 2012. Topological indoor Localization & Navigation for Autonomous Industrial Mobile Manipulator, Machine Learning and Application (ICMLA): 238.
5. FU Rong, JU Hehua. 2010. High-precision inverse kinematics based on discrete workspace for obstacle avoidance, Future Computer and Communication: 608.
6. Yahya, S. Mohamed, H.A.F. Moghavvemi, M. Yang, S.S. 2009. A new geometrical inverse kinematics method for planar hyper redundant manipulators, Innovative Technologies in Intelligent Systems and Industrial Applications: 20.
7. Srinivasan Alavandar, M. J. Nigam. 2008. Neuro-Fuzzy based Approach for Inverse Kinematics Solution of Industrial Robot Manipulators, International Journal of Computers, Communications & Control: 224.
8. Zhang Qingzhen, Liu Cunjia, Yang Bo, Ren Zhang. 2007. Re-entry Trajectory Planning Optimization Based on Ant Col-ony Algorithm, International Conference on Robotics and Biomimetics: 1064 – 1068.
9. Zhengxiong Liu, Panfeng Huang, Jie Yan. 2009. Multi-Objective Genetic Algorithms for Trajectory Optimization of Space Manipulator, Industrial Electronics and Applications: 2810.
10. Sang-Duck Lee, Byeong-Sang Kim, Jae-Bok Song. 2012. Guideline for Determination of Link Length of a 3 DOF Planar Manipulator for Human-Robot Collision Safety, Robotics and Automation (ICRA): 2685.
11. Chao Yu, Minghe Jin, Hong Liu. 2012. Analytical Solution For Inverse Kinematic Redundant Manipulators with Offset-Wrist, Mechatronics and Automation (ICMA): 92.
12. Sanghun Pyo, Syed Hassan, Yasir Jan, Jungwon Yoon. 2013. Design of 6-DOF Manipulator Intuitive Teaching System Using Smart Phone Orientation: User Friendly and Intuitive

- Teaching Operation for 6-DOF Manipulator, *Intelligent Systems Modelling & Simulation (ISMS)*: 364.
13. Dongsheng Guo, Yunong Zhang. 2012. New Inequality-Based Obstacle-Avoidance MVN Scheme and Its Application to Redundant Robot Manipulators, *Systems, Man, and Cybernetics, Part C: Application and Reviews*: 1326.
  14. Chen, C.-S. 2011. Supervisory adaptive tracking control of robot manipulators using interval type-2 TSK fuzzy logic systems, *Control Theory & Applications*: 1796.
  15. Zhang Bin, Xiong Rong, Wu Jun. 2011. Kinematics analysis of a novel 7-DOF humanoid manipulator for table tennis, *Electronics, Communications and Control (ICECC)*: 1524.
  16. Ho Pham Huy Anh, Nguyen Thanh Nam. 2012. Adaptive Trajectory Modelling of Humanoid Robot 3-DOF Arm Using Inverse Neural MIMO NARX Model: *Control, Automation and Information Sciences*: 381.
  17. Hsu-Chih Huang, Chien-Po Chen, Pei-Ru Wang. 2012. Particle Swarm Optimization for Solving the Inverse Kinematics of 7-DOF Robotic Manipulators, *Systems, Man, and Cybernetics (SMC)*: 3105.
  18. Nobuyuki Togashi, Tomoyuki Shimono, Naoki Motoi. 2012. Manipulability Servoing Control in Null Space for Redundant Bilateral Control System with Different Degrees of Freedom, *Advanced Motion Control (AMC)*: 1.
  19. Uwe Mettin, Pedro X. La Hera, Daniel Ortiz Morales, Anton S. Shiriaev, Leonid B. Freidovich, Simon Westerberg. 2009. Trajectory Planning and Time-Independed Motion Control for a Kinematically Redundant Hydraulic Manipulator, *Advanced Robotics*: 1.
  20. Wei Wang, Yan Li. 2009. Path Planning for Redundant Manipulator without Explicit Inverse Kinematics Solution, *Robotics and Biomimetics (ROBIO)*: 1918.
  21. G. Sartori Natal, A. Chemori, F. Pierrot. 2012. Dual-Space Adaptive Control of Redundantly Actuated Parallel Manipulators for Extremely Fast Operations With Load Changes, *Robotics and Automation (ICRA)*: 253
  22. Sang-Duck Lee, Byeong-Sang Kim, Jae-Bok Song. 2012. Guideline for determination of link length of a 3 DOF planar manipulator for human-robot collision safety, *Robotics and Automation (ICRA)*: 2685.
  23. Ali Morovati Pasand, Davood Naderi. 2013. Energy Optimization of a Planar Parallel Manipulator with Kinematically Redundant Degrees of Freedom, *Robotics and Mechatronics (ICRoM)*: 362.
  24. Paolo Righettini, Roberto Strada, Alberto Oldani, Andrea Ginamm. 2012. Experimental Activities on Step Motor Drives, 9th France-Japan & 7th Europe-Asia Congress on and Research and Education in Mechatronics (REM): 358.

25. XI Lei-ping, CHEN Zi-li, SUN Shao-jie. 2011. Obstacle Avoiding Research on the Manipulator based on Genetic Algorithm, Department of Optics and Electrics Engineering: 867.
26. M. Benzaoui, H. Chekireb, M. Tadjine. 2010. Redundant Robot Manipulator Control with Obstacle Avoidance Using Extended Jacobian Method, Controls & Automation (MED), 2010 18th Mediterranean Conference on: 371.
27. A. Taherifar, A. Alasty, H. Salarich, M. Boroushaki. 2013. Path Planning for a Hyper-Redundant Manipulator with Lockable Joints Using PSO, Robotics and Mechatronics (ICRoM): 224.
28. Christian Hansen, Julian Oltjen, David Meike, Tobias Ortmaier. 2012. Enhanced Approach for Energy-Efficient Trajectory Generation of Industrial Robots, Automation Science and Engineering (CASE): 1.
29. Francesca Ballan, Luca M. Capisani, Tullio Facchinetti, Antonella Ferrara, Alessandro Martinelli. 2010. Configuration space analysis oriented to robust control and obstacle avoidance of manipulators: Industrial Electronics (ISIE): 114.

## **PRIEDAI**

## **1. Mokslinės publikacijos**

1. Tomas Šarpnickas, Tomas Eglynas, Tomas Jankauskas, Jonas Vaupšas. Intelektualaus manipulatoriaus modelio programavimas ir optimalios trajektorijos parinkimas// VGTU: 14-oji Lietuvos jaunųjų mokslininkų konferencija „Mokslas – Lietuvos ateitis“, Vilnius, balandžio 15, 2011. ISBN 978-9955-28-834-3.I.
2. Šarpnickas, T.; Bulbenkienė, V. Manipulatoriaus išmaniosios valdymo sistemos kūrimas ir energetinių kaštų minimizavimas// KU: Technologijos mokslo darbai vakarų Lietuvoje VIII, Klaipėda, gegužės 18, 2012. ISSN 1822-4652.

## INTELEKTUALAUS MANIPULATORIAUS MODELIO PROGRAMAVIMAS IR OPTIMALIOS TRAJEKTORIJOS PARINKIMAS

Tomas Šarpanickas<sup>1</sup>, Tomas Eglynas<sup>2</sup>, Tomas Jankauskas<sup>3</sup>, Jonas Vaupšas<sup>4</sup>

Klaipėdos universitetas

El. paštas: <sup>1</sup>sarpanickas.tomas@gmail.com; <sup>2</sup>tmse@inbox.lt; <sup>3</sup>rootax@gmail.com; <sup>4</sup>vaupsas@gmail.com

**Santrauka.** Nagrinėjant manipulatoriaus atvirkštinės kinematikos uždavinius, būtina apskaičiuoti lanksčiai sujungtų svirčių trajektorijas, atsižvelgiant į objekto koordinatėjų kitimo dėsnį. Eksperimentiniam tyrimui sukonstruoto ir pagaminto manipulatoriaus modelio kinematikos lygčių užrašymas ir jų analizė suteikia galimybę sudaryti programavimo algoritmą svirčių kampų skaičiavimui ir judančiųjų dalių trajektorijos optimizavimui, siekiant sumažinti kelio ilgį ir efektyviau panaudoti manipulatoriaus variklius.

**Reikšminiai žodžiai:** manipulatorius, atvirkštinė kinematika, trajektorijos optimizavimas, žingsniniai varikliai.

### Įvadas

Įvairių konstrukcijų manevringi, intelektualūs ir tikslūs manipulatoriai, gebantys patys tinkamai koordinuoti savo veiksmus, yra plačiai taikomi daugelyje šiuolaikinės gamybos ir mokslo sričių (Hao Meng *et al.* 2007; Fuhai Zhang *et al.* 2010).

Viena iš svarbiausių manipulatoriaus kūrimo problemų yra siejama su aktualia atvirkštinės kinematikos uždaviniu - lanksčiai sujungtų svirčių kampų skaičiavimu, norint tinkamai valdyti tų svirčių padėtį erdvėje pagal nustatytas darbo įrankio koordinates ir jo pozicionavimo sąlygas. (FU Rong *et al.* 2010; Yahya *et al.* 2009). Kita ne mažiau svarbi problema yra susijusi su judančiųjų dalių optimalios trajektorijos parinkimu, nuo kurio priklauso sinchronizuotas ir energetiškai efektyvus manipulatoriaus variklių darbas (Zhang Qingzhen *et al.* 2007; Zhengxiong Liu *et al.* 2009).

Siekiant išnagrinėti šių abiejų uždavinių sprendimo galimybes, buvo sukonstruotas ir pagamintas medžio rąstų krovimo įrenginio manipulatoriaus eksperimentinis modelis (1 pav.), skirtas svirčių kinematikos matematinei analizei atlikti ir optimalios trajektorijos valdymo algoritmo sudaryti.

### Manipulatoriaus modelio įranga ir funkcijos

Manipulatoriaus modelio (1 pav.) konstrukcija ir svirčių kinematika atitinka pramoninio medžio rąstų krovimo roboto įrangos principus ir funkcijas. Tai įrenginys, galintis atlikti sudėtingą erdvinį judesį,

imituojantį žmogaus rankos veiksmus (Yongseon Moon *et al.* 2007).

Manipulatoriaus modelyje yra įmontuoti penki žingsniniai varikliai, kurie 2 pav. yra atitinkamai pažymėti  $M_1, \dots, M_5$ . Žingsninių variklių pavarose yra įrengti mechaniniai reduktoriai, kurie leidžia sumažinti modelio svirčių sukimosi greitį ir pasiekti aukštą pozicionavimo tikslumą. Svarbiausią funkciją –



svirčių judesių koordinavimą ir pozicionavimą - atlieka trys varikliai -  $M_1, M_2$  ir  $M_3$ .

1 pav. Medžio rąstų krovimo manipulatoriaus modelis

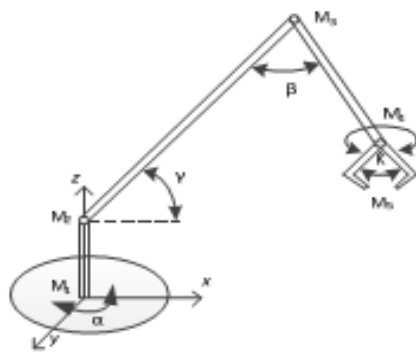
Kaip matome iš 2 pav., nuo svirčių pasisukimo kampų  $\alpha, \beta$  ir  $\gamma$  priklauso manipulatoriaus pilnutinai atverto griebtuvo padėtį atitinkančio taško K koordinatės  $x, y$  ir  $z$ . Taigi, manipulatoriaus variklio  $M_1$  sukamos svirties kampinė koordinatė  $\alpha$  nustato taško K padėtį  $x, y$  koordinatėjų plokštumoje, o variklių  $M_2$  ir  $M_3$  varomų svirčių kampai  $\beta$  ir  $\gamma$  lemia griebtuvo

taško K padėtį visose trijose koordinatinių ašyse. Todėl tam, kad darbo metu manipulatoriaus griebtuvas atsiras būtų reikalingoje padėtyje, reikia žinoti ne tik svirčių kampines padėtis prieš judėjimo pradžią, bet ir tų svirčių galutines kampines koordinates, kurios tiksliai atitiktų reikalingą griebtuvo taško K padėtį erdvėje.

Taigi, siekiant suderinti žingsninių variklių valdymą, yra būtina sudaryti lygtis, aprašančias taško K judėjimo dėsningumus atitinkamoje koordinatinių sistemoje ir sudaryti manipulatoriaus svirčių pasisukimo kampų skaičiavimo programos algoritmą.

### Svirčių kampinės padėties skaičiavimas

Lankščiai sujungtų manipulatoriaus svirčių kampinių koordinatinių padėčių skaičiavimas yra atliekamas, pagal griebtuvo taško K (objekto) koordinates, atsižvelgiant į manipulatoriaus konstrukciją ir jo svirčių bei žingsninių variklių išdėstymo schemą (2 pav.). Tai leidžia nustatyti visų manipulatoriaus variklių rotorų



2 pav. Žingsninių variklių išdėstymo ir svirčių eigos schema

sukimosi kryptį, jų posūkio kampų vertes ir, reikalui esant, variklių valdymo impulsų skaičių ir parametrus.

Atsižvelgiant į manipulatoriaus konstrukcijos technines galimybes, objekto koordinatinių kitimas yra apribojamas šiomis sąlygomis:

$$\begin{aligned} z &\geq 0; \\ \sqrt{x^2 + y^2 + z^2} &\leq A + B. \end{aligned} \quad (1)$$

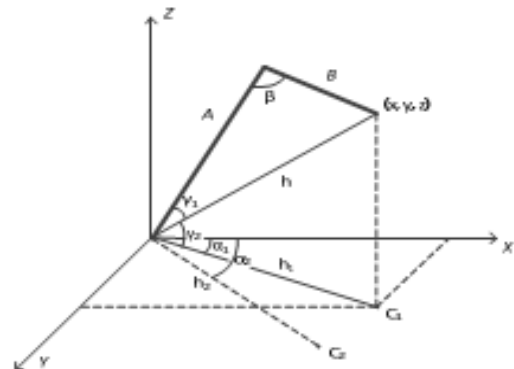
kur  $A, B$  – manipulatoriaus svirčių ilgiai (3 pav.).

**Kampo  $\alpha$  radimas.** Manipulatoriaus pirmojo variklio  $M_1$  reguliuojamas įrenginio posūkio kampas  $\alpha$  yra apskaičiuojamas pagal reikalingas objekto padėties koordinates  $x, y, z$ . Tačiau šio kampo skaičiavimo metodas, kaip yra nurodoma 1 lentelėje, priklauso nuo pasirinktųjų koordinatinių  $x$  ir  $y$  konkrečios vertės.

1 lentelė. Kampo  $\alpha$  skaičiavimas pagal koordinatinių reikšmes

$\alpha$	$x < 0$	$x = 0$	$x > 0$
$y < 0$	$\tan^{-1} \frac{y}{x} - 180^\circ$	$-90^\circ$	$\tan^{-1} \frac{y}{x}$
$y = 0$	$180^\circ$	-	$0^\circ$
$y > 0$	$\tan^{-1} \frac{y}{ x } + 90^\circ$	$90^\circ$	$\tan^{-1} \frac{y}{x}$

Kaip matome, 1 lentelėje yra pateikta arba kampo  $\alpha$  skaičiavimo formulė, arba šio kampo konkreti vertė, įrenginiui esant tarp teigiamųjų koordinatės  $x$  verčių ašies (3 pav.) ir tiesės  $h_1$   $[(0,0),(x,y)]$ . Taigi, viso įrenginio sukimo kampo  $\alpha$  vertės gali būti apibrėžtame intervale  $-180^\circ < \alpha \leq 180^\circ$ . Jei kampas  $\alpha$  yra neigiamas ( $\alpha < 0$ ), tai manipulatorius yra sukamas laikrodžio rodyklės kryptimi, o jei kampas  $\alpha$  yra teigiamas ( $\alpha > 0$ ) – priešinga kryptimi. Todėl, norint apskaičiuoti, kokių kampu reikia pasukti objektą nuo taško  $C_1$  iki taško  $C_2$ , reikia žinoti kampų  $\alpha_1$  (nuo teigiamosios  $x$  ašies dalies iki tiesės  $h_1$ ) ir kampo  $\alpha_2$  (nuo teigiamosios  $x$  ašies dalies iki tiesės  $h_2$ ) vertes.



3 pav. Modelio svirčių kampinė padėtis koordinatinių sistemoje

Posūkio kampo  $\alpha$  vertė ir trumpiausias griebtuvo kelias, manipulatoriui pasisukant tarp minėtųjų dviejų taškų, yra apskaičiuojamas, atsižvelgiant į sąlygas:

- kai  $|a_2 - a_1| \leq 180^\circ$ , tai  $a = a_2 - a_1$
- kai  $|a_2 - a_1| \geq 180^\circ$ , tai  $a = -(360^\circ - |a_2 - a_1|)$

**Kampų  $\gamma$  ir  $\beta$  radimas.** Kampai  $\gamma$  ir  $\beta$  gali būti apskaičiuojami pagal svirčių  $A$  ir  $B$  savitarpės padėties trikampį (3 pav.), susidarantį atkarpos  $A$  ir  $B$  sujungus tiese  $h$ , kurios ilgis yra nustatomas pagal formulę:

$$h = \sqrt{x^2 + y^2 + z^2} \quad (2)$$

Kampai  $\beta$  ir  $\gamma$  apskaičiuojami iš formulių:

$$\beta = \cos^{-1} \frac{B^2 + A^2 - x^2 - y^2 - z^2}{2AB} \quad (3)$$

$$\gamma = \gamma_1 + \gamma_2 = \cos^{-1} \frac{A^2 - B^2 + x^2 + y^2 + z^2}{2A\sqrt{x^2 + y^2 + z^2}} + \sin^{-1} \frac{z}{\sqrt{x^2 + y^2 + z^2}} \quad (4)$$

#### Variklių darbo laiko suderinimas, taikant linijinį programavimą

Siekiant sutrumpinti manipulatoriaus variklių darbo laiką ir tuo būdu padidinti viso įrenginio našumą, turi būti išspręstas griebtuvo judėjimo trajektorijos optimizavimo uždavinys. Griebtuvo taško K trajektorijos ilgio mažinimas gali būti atliekamas atitinkamai derinant žingsninių variklių darbo trukmę. Šiam tikslui taikant linijinį programavimą, galima suformuoti variklių valdymo impulsų užlaikymams siunčiamus optimalius signalus, kartu išlaikant maksimalų ilgiausiai turinčio dirbti variklio greitį. Šis maksimizavimas gali būti atliekamas pagal funkciją -

$$F_{\max} = \varphi_1 \cdot \omega_1 + \varphi_2 \cdot \omega_2 + \varphi_3 \cdot \omega_3, \quad (5)$$

kur:  $\varphi$  - redukuotasis kampas, išreiškiamas vienam svirties posūkio laipsniui reikalingo impulsų skaičiaus ir svirties posūkio kampo sandauga,

$\omega$  - žingsninio variklio impulsų dažnis.

Šio uždavinio sprendimui yra sudaromos lygtys:

$$\varphi_1 \cdot \omega_1 - \varphi_2 \cdot \omega_2 = 0, \quad (6)$$

$$\varphi_1 \cdot \omega_1 - \varphi_3 \cdot \omega_3 = 0. \quad (7)$$

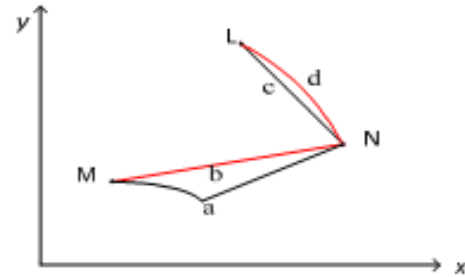
Pagal manipulatoriaus žingsninių variklių specifikacijas, impulsų dažnis neturėtų viršyti 500, todėl:

$$\omega_1, \omega_2, \omega_3 \leq 500. \quad (8)$$

Didžiausioji funkcijos  $F_{\max}$  vertė, kuri turi atitikti nustatytas sąlygas (11)-(13), apibrėžia ir didžiausias visų šios funkcijos dėmenų vertes. Todėl griebtuvui artėjant į galutinę padėtį, sistema veiks sparčiausiai, kai maksimaliu greičiu suksis ilgiausiai turintis dirbti variklis ir kai prie jo bus derinama ir kitų variklių darbo trukmė.

Taigi, pagal funkcijos  $F_{\max}$  maksimizavimo sąlygas nustačius redukuotųjų kampų vertes  $\varphi_1, \varphi_2, \varphi_3$ , atitinkančias reikalingą svirčių padėtį, galime apskaičiuoti kiekvieno variklio impulsų dažnius ir tuo būdu suderinti jų darbo trukmę.

Manipulatoriaus kinematikos analizė rodo, jog yra galimi keli skirtingi griebtuvo judėjimo trajektorijos iš taško M į N (5 pav.) atvejai.



4 pav. Taško K trajektorijos projekcija į x, y plokštumą

Trajektorijos projekcija a (5 pav.) atitinka atvejį, kai visi varikliai yra įjungiami vienu metu, tačiau dėl griebtuvo judesiu reikalingo skirtingo veikimo laiko, jie sustoja ne vienu metu. Tokia trajektorija nėra racionali dėl ilgo kelio ir dėl staigaus judėjimo krypties keitimo, kuris didina įrenginio apkrovą ir kenkia konstrukcijos stabilumui.

Trajektorijos projekcija b atitinka atvejį, kai visi varikliai įsijungia ir išsijungia vienu metu. Šiuo atveju varikliai turi veikti skirtingais greičiais, ir tai leidžia pasiekti galutinį tašką optimaliai trumpiausia trajektorija.

Suderinus variklių darbo trukmę, griebtuvas gali judėti optimaliausia kreive d, nes, tiesės c atveju, įrenginio variklius pirmiausiai reikia sukti į vieną pusę, o vėliau - į priešingą pusę, tuo būdu atliekant didesnę darbą. Taigi, šiuo atveju galima sumažinti variklių energijos sąnaudas, nepaisant tam tikro trajektorijos pa ilgėjimo.

Taigi, synchronizavus žingsninių variklių veikimo laikus, galimi du trajektorijos optimizavimo atvejai, tai tiesiaieigė b ir kreivaeigė d trajektorijos (5 pav.), tačiau abu yra optimaliausi manipulatoriaus griebtuvo judėjimo trajektorijos įmanomi variantai.

#### Programos algoritmas

Variklių  $M_1, M_2$  ir  $M_3$  valdomų svirčių pasisukimo kampų skaičiavimo programos kodo algoritmas yra pavaizduotas 6 pav. Dėl sudėtingų skaičiavimų, vykstančių algoritmo viduje, yra sudaromas konkretus išorinis proceso algoritmas, atitinkantis pagrindinius programos etapus.

Skaičiavimo procesui reikalingas pradinės manipulatoriaus griebtuvo koordinatės gali įvesti arba pats operatorius, arba šių koordinatė vertės į mikrovaldiklį gali būti perduodamos iš įrenginio jutiklių. Sistema, patikrindama koordinatė atitikimą nustatytoms pradinėms sąlygoms, skaičiavimus atlieka pagal formulę

(1). Jei koordinatinių vertės atitinka sistemos technines galimybes, tai programa atlieka visus kampų skaičiavimus optimalaus kelio nustatymui arba skaičiavimus grąžina į proceso pradžią. Atlikus kampų ir sinchronizavimo sąlygų skaičiavimą, žingsniniams varikliams yra perduodamos atitinkamai parengtos vykdymo komandos, kuriose yra koduojama informacija apie variklių rotorių kampines padėtis bei užlaikymo signalus.



5 pav. Skaičiavimų ir proceso valdymo algoritmas

Pagal šį algoritmą yra parašytas programos kodas, kuris susideda iš kelių funkcijų:

- iš pagrindinės funkcijos, kurioje yra įvedamos koordinatinių vertės ir kuri kreipiasi į kitas funkcijas ir iš jų gauna informaciją bei ją teikia kitoms sub-funkcijoms;
- iš koordinatinių verifikavimo funkcijos, kuri tikrina pagrindinės funkcijos perduotas koordinates ir, jei jos atitinka nustatytąsias sąlygas, tai informacija apie koordinates yra grąžinama į pagrindinę funkciją, jei neatitinka - yra nukreipiama pakartotiniam įvedimui;
- kampų skaičiavimo funkcijos, kuri pagal teisingai įvestas koordinates ir integruotas formules apskaičiuoja manipulatoriaus svirčių kampus ir skaičiavimo rezultatus grąžina pagrindinei funkcijai;
- optimalaus kelio parinkimo funkcijos, kuri pagal gautąsias kampų vertes apskaičiuoja žingsninių variklių impulsų skaičių ir užlaikymo trukmes ir tada šiuos duomenis grąžina pagrindinei funkcijai;
- variklių paleidimo funkcijos, kuri pagal žingsninių variklių impulsų skaičius ir užlaikymo trukmes šiuos variklius paleidžia ir valdo jų darbą.

## Išvados

- Manipulatoriaus modelio svirčių padėties matematinis įvertinimas ir teorinė analizė leidžia sudaryti svirčių kampinių koordinatinių skaičiavimo lygtis ir suformuoti optimalios griebtuvo judėjimo trajektorijos sąlygas.
- Eksperimentinio manipulatoriaus modelio energetinis efektyvumas gali būti padidintas, sinchronizuojant žingsninių variklių darbą pagal sudarytą svirčių kampų skaičiavimo programos algoritmą ir koduotas variklių rotorių kampinės padėties nustatymo komandas bei užlaikymo signalus.

## Literatūra

- Hao Meng, Lina Cao, Yong Zhang. 2007. Posture Stabilization of Manipulator Based on Rotation Platform, International Conference on Mechatronics and Automation: 88 – 93.
- Fuhai Zhang, Yili Fu, Shibo Yan, and Shuguo Wang. 2010. Kinematics and simulations of space manipulator in three work modes, Systems and Control in Aeronautics and Astronautics: 572 – 577.
- FU Rong, JU Hehua. 2010. High-precision inverse kinematics based on discrete workspace for obstacle avoidance, Future Computer and Communication: 608 – 613.
- Yahya, S. Mohamed, H.A.F. Moghavvemi, M. Yang, S.S. 2009. A new geometrical inverse kinematics method for planar hyper redundant manipulators, Innovative Technologies in Intelligent Systems and Industrial Applications: 20 – 22.
- Zhang Qingzhen, Liu Cunjia, Yang Bo, Ren Zhang. 2007. Re-entry Trajectory Planning Optimization Based on Ant Colony Algorithm, International Conference on Robotics and Biomimetics: 1064 – 1068.
- Zhengxiong Liu, Panfeng Huang, Jie Yan. 2009. Multi-Objective Genetic Algorithms for Trajectory Optimization of Space Manipulator, Industrial Electronics and Applications: 2810 – 2815.
- Yongseon Moon, Jinju Park, Cheol-Ho Yun, Nak Yong Ko, Hong Sik Moon, Youngchul Bae. 2007. The Design of humanoid Robot Arm based on Morphological and Neurological Analysis of Human Arm, Industrial Electronics Society: 2760 – 2765.

## PROGRAMMING AND THE OPTIMAL TRAJECTORY PREFERENCE OF AN INTELLIGENT MANIPULATOR MODEL

T. Šarpićas, T. Eglynas, T. Jankauskas, J. Vaupšas

### Abstract

Analysis of manipulator inverse kinematics needs for compiling of arm movement equations which allow specifying requirements with the view of the optimized trajectory and more effective usage of stepper motors. The thoughtful research enabled to create an algorithm for computation of angles of the arms by taking into ac-

count the conditions determined by experimental model of timber log manipulator.

**Keywords:** manipulator, inverse kinematics, trajectory optimization, stepper motors.

# Manipulatoriaus išmaniosios valdymo sistemos kūrimas ir energetinių kaštų minimizavimas

T. Šarpnickas, V. Bulbenkienė

Klaipėdos Universitetas, Informatikos inžinerijos katedra, Bijūnų 19, 91225 Klaipėda, Lietuva, E-mail: sarpnickas.tomas@gmail.com, bulbenkiene@ik.ku.lt

## Anotacija

Išmanūs ir manevringi manipulatoriai, gebantys patys tinkamai koordinuoti savo veiksmus, yra plačiai taikomi didžiuosiuose pasaulio uostuose, geležinkeliuose bei įvairiose gamybos ir mokslo srityse. Nagrinėjant manipulatoriaus atvirkštinės kinematikos uždavinius, būtina apskaičiuoti lanksčiai sujungtų svirčių trajektorijas, atsižvelgiant į objekto koordinatinių kitimo dėsnį. Manipulatoriaus modelio kinematikos lygčių užrašymas ir jų analizė suteikia galimybę sudaryti programos algoritmą svirčių kampų skaičiavimui ir judančiųjų dalių trajektorijos optimizavimui, siekiant sumažinti kelio ilgį ir efektyviau panaudoti manipulatoriaus variklius, tuo pačiu sumažinti energijos sąnaudas. Šiame darbe analizuojama manipulatoriaus kinematika bei pateikiamas algoritmas judančių dalių trajektorijų optimizavimui, siekiant sumažinti energetinius kaštus.

**Reikšminiai žodžiai:** išmaniosios valdymo sistemos, manipulatorius, atvirkštinės kinematikos uždavinys.

## Abstract

Smart and manoeuvrable manipulators, which are able to decide their movements, are widely applied in the world's biggest ports, railway stations and in manufacturing and science. Analysis of manipulator inverse kinematics needs for compiling of arm movement equations which allow specifying requirements with the view of the optimized trajectory and more effective usage of motors. The thoughtful research enables to create an algorithm for computation of angles of the arms by taking into account the conditions determined by model of manipulator. In this paper inverse kinematics of manipulator is analysed and algorithm is created for trajectory optimization to reduce energy costs.

**Key words:** smart control systems, manipulator, inverse kinematics.

## 1. Įvadas

Manipuliatorių galima apibrėžti kaip įrenginį, kuris gali atlikti sudėtingą erdvinį judesį, imituojantį žmogaus rankos veiksmus [1]. Įvairių konstrukcijų manevringi, intelektualūs ir tikslūs manipulatoriai, gebantys patys tinkamai koordinuoti savo veiksmus, yra plačiai taikomi uostuose, geležinkeliuose bei įvairiose gamybos ir mokslo srityse [2].

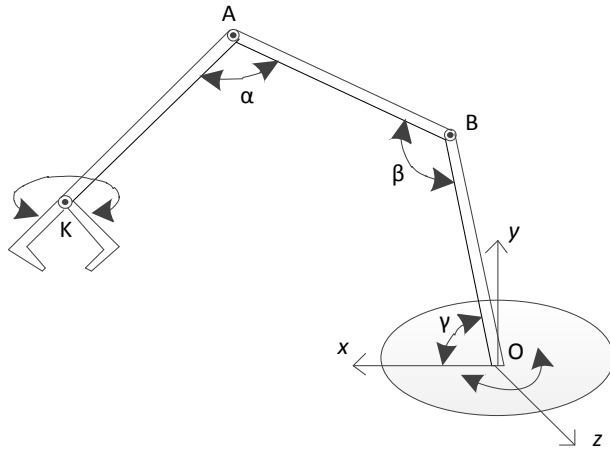
Siekiant užtikrinti efektyvų sistemų valdymą, ypač svarbu yra tinkamai aprašyti sistemos veikimo algoritmą, leidžiantį sukurti išmaniosios autonominės sistemos valdomą manipuliatorių, kuris būtų taikomas įvairiose pramonės srityse [3]. Viena iš svarbiausių manipulatoriaus kūrimo problemų yra siejama su aktualia atvirkštinės kinematikos uždaviniu - lanksčiai sujungtų svirčių kampų skaičiavimu, norint tinkamai valdyti tų svirčių padėčių erdvėje pagal nustatytas darbo įrankio koordinatas ir jo pozicionavimo sąlygas [4]. Kita ne mažiau svarbi problema yra susijusi su svirčių tarpusavės padėties kampų konfigūracijos parinkimu, nuo kurio priklauso energetiškai efektyvus manipulatoriaus variklių darbas [5].

Nagrinėjant šių abiejų uždavinių sprendimo galimybes, buvo suprojektuotas manipulatoriaus eksperimentinis modelis, skirtas svirčių kinematikos matematinei analizei atlikti ir optimaliam valdymo algoritmui sudaryti.

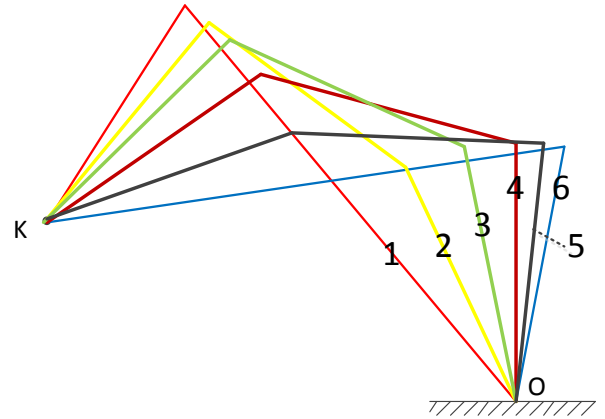
## 2. Manipulatoriaus išmanioji valdymo sistema

Darbe analizuojama manipulatoriaus išmanioji valdymo sistema. Šiuo atveju nagrinėjamas trijų svirčių manipulatoriaus modelis (1 pav.). Esant trimis svirtims ir daugiau gali būti sudaryta begalybė galimų svirčių kampų konfigūracijų užduočiai atlikti. 2 paveiksle pavaizduota keletas svirčių padėčių variantų manipulatoriaus griebtuvui esant taške K. Siekiant sumažinti variklių energetines sąnaudas, labai svarbu parinkti optimaliausias manipulatoriaus svirčių tarpusavės padėtis.

Manipulatoriaus valdymo sistema įsimeina griebtuvo (K taško) koordinatas ir svirčių kampines padėtis po paskutinės užduoties atlikimo. Pagal užduotas naujas objekto, kurį turi pasiekti griebtuvas, koordinatas sistema apskaičiuoja kokiu kampu atitinkamai pasukti kiekvieną svirtį nuo savo pradinių padėčių, kad būtų minimizuoti energetiniai kaštai: visų variklių bendras energijos suvartojimas būtų minimalus.



1 pav. Manipulatoriaus modelis



2 pav. Svirčių tarpusavės padėties variantai

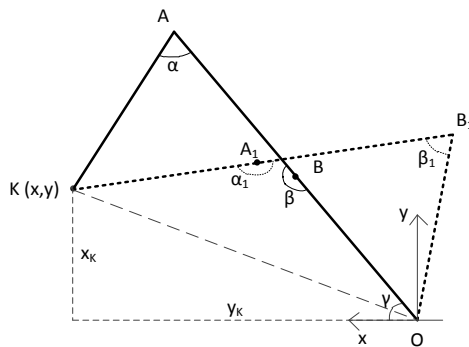
Taigi, siekiant minimizuoti variklių energetinius kaštus, yra būtina sudaryti lygtis, pagal taško K koordinatės apskaičiuojančias svirčių tarpusavės padėties kampų konfigūracijas. Konfigūracijų kiekis apribojamas įvedus kampo posūkio žingsnį, nagrinėjamu atveju  $0,001^0$ . Manipulatoriaus svirčių judesių koordinavimas aprašomas kampais  $\alpha$ ,  $\beta$ ,  $\gamma$  (1 pav.) ir jų vertės yra intervale  $0^0 - 180^0$ .

Žinant objekto koordinatės yra paskaičiuojami visi įmanomi manipulatoriaus svirčių padėties kampai  $0,001^0$  žingsniu, kurie suvedami į lentelę. „Matlab“ programos pagalba Lentelėje esantys kampai lyginami su pradinės padėties kampais ir parenkamas mažiausiai energetinių resursų naudojantis variantas griebtuvui pasiekti objektą.

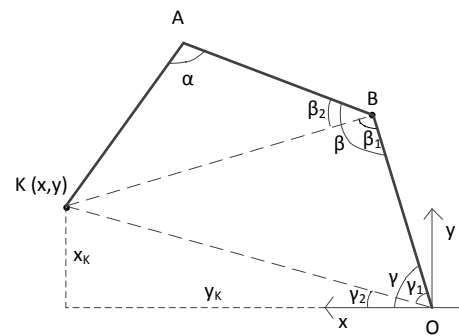
### 3. Svirčių kampinės padėties skaičiavimas

Iš pradžių randamas pasirinkto kampo (šiuo atveju -  $\alpha$ ) intervalas pagal objekto koordinatės. Sudaroma kampų  $\alpha$  aibė, kurioje yra visi kampai nuo minimalios  $\alpha$  padėties iki maksimalios besiskiriantys žingsniu -  $0,001^0$ . Išvedamos  $\beta$ ,  $\gamma$  kampų skaičiavimo formulės, kuriose vertės paskaičiuojamos prie kiekvienos  $\alpha$  aibės reikšmės.

Svirtys yra vienodo ilgio, todėl  $KA = AB = BO = OB_1 = B_1A_1 = A_1K = L$ . Objektas negali būti toliau negu  $3L$  (pilnai ištiestos manipulatoriaus svirtys) ir ne arčiau nei per vieną svirtį ( $KO > L$ ) nuo manipulatoriaus įtvirtinimo prie pagrindo. Kampo  $\alpha$  intervalai skaičiuojami dviem būdais, priklausomai nuo taško K koordinatė.



3 pav. Manipulatoriaus ribiniai kampai



4 pav. Manipulatoriaus kampinės padėtys

Tikriname ar  $3L > KO > L$ . Jei nelygybė teisinga, tai kampas  $\alpha$  gali įgyti mažiausią įmanomą vertę, kai svirtys sudarys trikampį AOK, tada  $\beta$  įgis maksimalią vertę -  $180^0$ . Pagal formulę (1) randamas KO, o L yra žinomas (svirčių ilgis). Kampo  $\alpha$  minimali vertė skaičiuojama naudojant kosinusų teorema (3 pav.):

$$KO = \sqrt{x_k^2 + y_k^2}; \quad (1)$$

$$KO^2 = L^2 + (2L)^2 - 2L(2L) \cos \alpha; \quad (2)$$

$$\alpha = \cos^{-1} \left( \frac{5L^2 - (x_k^2 + y_k^2)}{4L^2} \right). \quad (3)$$

Naudojant kosinusų teoremą (4 pav.) per kampus  $\alpha$  ir  $\gamma_1$  yra išreiškiamas  $KB^2$ . Sulyginus (4) ir (5) formulė dešines puses išreiškiamas kampas  $\gamma_1$  ir iš (1) formulės įstatoma KO išraiška:

$$KB^2 = L^2 + L^2 - 2L^2 \cos \alpha; \quad (4)$$

$$KB^2 = KO^2 + L^2 - 2KO \cdot L \cos \gamma_1; \quad (5)$$

$$\gamma_1 = \cos^{-1} \frac{x_k^2 + y_k^2 - L^2(1 - 2 \cos \alpha)}{2L \sqrt{x_k^2 + y_k^2}}. \quad (6)$$

Turint kampą  $\gamma_1$  (4 pav.) paskaičiuojamas kampas  $\gamma$ :

$$\gamma = \gamma_1 + \gamma_2; \quad (7)$$

$$\gamma_2 = \tan^{-1} \frac{x_k}{y_k}; \quad (8)$$

$$\gamma = \cos^{-1} \frac{x_k^2 + y_k^2 - L^2(1 - 2 \cos \alpha)}{2L \sqrt{x_k^2 + y_k^2}} + \tan^{-1} \frac{x_k}{y_k}. \quad (9)$$

$\beta$  išreiškiamas kaip  $\beta_1$  ir  $\beta_2$  suma (4 pav.). Pagal kosinusų teoremą išreiškiamas kampas  $\beta_1$ . Trikampis  $ABK$  lygiašonis, todėl kampas  $\beta_2$  ir kampas  $AKB$  yra lygūs ir išreiškiamas pagal (13) formulę. (14) formulėje sudėjus  $\beta_1$  ir  $\beta_2$  vietoj  $KB^2$  įkelta formulė (5):

$$\beta = \beta_1 + \beta_2; \quad (10)$$

$$KO^2 = KB^2 + L^2 - 2KB \cdot L \cos \beta_1; \quad (11)$$

$$\beta_1 = \cos^{-1} \frac{KB^2 + L^2 - KO^2}{2KB \cdot L}; \quad (12)$$

$$\beta_2 = \frac{180^\circ - \alpha}{2}; \quad (13)$$

$$\beta = \cos^{-1} \frac{L - KO \cos \gamma_1}{\sqrt{KO^2 + L^2 - 2KO \cdot L \cos \gamma_1}} + \frac{180^\circ - \alpha}{2}. \quad (14)$$

#### 4. Kampai aprašantys manipulatoriaus svirčių judesių koordinavimą

Pasinaudojus (3), (9) ir (14) formulėmis galima sudaryti manipulatoriaus svirčių judesių koordinavimą aprašančių kampų  $\beta$ ,  $\gamma$  kitimą, kampui  $\alpha$  kintant  $0,001^0$  žingsniu. Gauti kampai pateikti lentelėje 1.

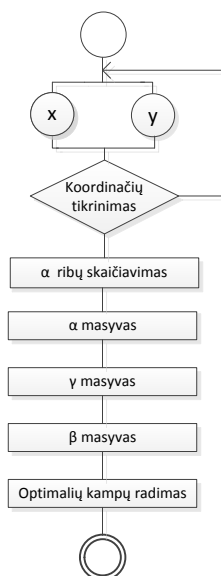
Lentelė 7. Manipulatoriaus svirčių judesių koordinavimą aprašantys kampai

Nr.	Kampas $\alpha$	Kampas $\gamma$	Kampas $\beta$
1.	$\cos^{-1} \left( \frac{5L^2 - (x_k^2 + y_k^2)}{4L^2} \right)$	$\cos^{-1} \frac{x_k^2 + y_k^2 - L^2(1 - 2 \cos \alpha)}{2L \sqrt{x_k^2 + y_k^2}} + \tan^{-1} \frac{x_k}{y_k}$	$\cos^{-1} \frac{L - KO \cos \gamma_1}{\sqrt{KO^2 + L^2 - 2KO \cdot L \cos \gamma_1}} + \frac{180^\circ - \alpha}{2}$
2.	$\cos^{-1} \left( \frac{5L^2 - (x_k^2 + y_k^2)}{4L^2} \right) + 0,001^0$	$\cos^{-1} \frac{x_k^2 + y_k^2 - L^2(1 - 2 \cos \alpha)}{2L \sqrt{x_k^2 + y_k^2}} + \tan^{-1} \frac{x_k}{y_k}$	$\cos^{-1} \frac{L - KO \cos \gamma_1}{\sqrt{KO^2 + L^2 - 2KO \cdot L \cos \gamma_1}} + \frac{180^\circ - \alpha}{2}$
3.	$\cos^{-1} \left( \frac{5L^2 - (x_k^2 + y_k^2)}{4L^2} \right) + 0,002^0$	$\cos^{-1} \frac{x_k^2 + y_k^2 - L^2(1 - 2 \cos \alpha)}{2L \sqrt{x_k^2 + y_k^2}} + \tan^{-1} \frac{x_k}{y_k}$	$\cos^{-1} \frac{L - KO \cos \gamma_1}{\sqrt{KO^2 + L^2 - 2KO \cdot L \cos \gamma_1}} + \frac{180^\circ - \alpha}{2}$
4.	...	...	...
...	$180^0$	$\cos^{-1} \frac{3(x_k^2 + y_k^2 - L^2)}{2L \sqrt{x_k^2 + y_k^2}} + \tan^{-1} \frac{x_k}{y_k}$	$\cos^{-1} \frac{L - KO \cos \gamma_1}{\sqrt{KO^2 + L^2 - 2KO \cdot L \cos \gamma_1}}$

Ši sudaryta lentelė leidžia, pasinaudojus „Matlab“ programa, parinkti optimaliausią manipulatoriaus svirčių kampinės padėties variantą, atsižvelgiant į atitinkamų kampų reikiamą pokytį užduočiai atlikti. Manipulatoriaus pradinės būsenos kampai  $\alpha$ ,  $\beta$ ,  $\gamma$  „Matlab“ programoje yra lyginami su kiekviena eilute iš lentelės 1 ir yra išrenkami kampai, kurie mažiausiai skiriasi nuo pradinių kampų.

#### 5. Išmaniosios valdymo sistemos algoritmas

Manipulatorius įsima paskutinės atliktos užduoties  $K$  taško koordinatės ir svirčių tarpusavės padėties kampus (savo esamą būseną). Operatorius įveda objekto koordinatės arba jos yra nuskaitomos jutiklių pagalba (5 pav.). Tikrinama ar koordinatės patenka į manipulatoriaus darbingumo zoną ( $3L > KO > L$ ). „Matlab“ programoje sudaromas  $\alpha$  kampų masyvas, kur  $\alpha$  kinta  $0,001^0$  žingsniu. Pagal  $\alpha$  kampus atitinkamai yra paskaičiuojami  $\gamma$  kampai. Turint  $\alpha$ ,  $\gamma$  kampus paskaičiuojami  $\beta$  kampai. Toliau yra atliekamas optimalių kampų radimas, kuris yra aprašytas 4 skyriuje.



5 pav. Sistemos algoritmas

## 6. Išvados

Manipulatoriaus svirčių kampinės padėties teorinė analizė leidžia sudaryti svirčių tarpusavės padėties kampų skaičiavimo lygtis ir parinkti optimalias svirčių padėtis.

Manipulatoriaus variklių energetinis efektyvumas gali būti padidintas, optimaliai parinkus svirčių kampines padėtis pagal sudarytą svirčių kampų skaičiavimo bei mažiausio svirčių tarpusavės padėties kampų kitimo radimo algoritmą.

## Literatūra

1. Bin Zhang, Rong Xiong, Jim Wu, 2011. Kinematics and trajectory planning of a novel humanoid manipulator for table tennis. *Electrical and Control Engineerin*), International Conference ICECE, pp. 3047 – 3051.
2. Fuhai Zhang, Yili Fu, Shibo Yan, and Shuguo Wang. 2010. Kinematics and simulations of space manipulator in three work modes. *Systems and Control in Aeronautics and As-tronautics*, 3rd International Symposium (ISSCAA), pp. 572 – 577.
3. Šarpnickas T., Eglynas T., Janakauskas T., Vaupšas J. 2011. Intelektualaus manipulatoriaus modelio programavimas ir optimalios trajektorijos parinkimas, *Mokslas - Lietuvos ateitis*, p. 4. <http://dspace.vgtu.lt/handle/1/759>.
4. FU Rong, JU Hehua. 2010. High-precision inverse kinematics based on discrete workspace for obstacle avoidance. *Future Computer and Communication*. 2nd International Conference ICFCC, pp. 608 – 613.
5. Zhengxiong Liu, Panfeng Huang, Jie Yan. 2009. Multi-Objective Genetic Algorithms for Trajectory Optimization of Space Manipulator. *Industrial Electronics and Applications*. 4th IEEE Conference ICIEA, pp. 2810 – 2815.

## **2. Kompaktinė plokštelė**

1. Tomas\_Šarpnickas\_TISI\_Magistro\_Baigiamasis\_Darbas.docx
2. Tomas\_Šarpnickas\_TISI\_Magistro\_Baigiamojo\_darbo\_prezentacija.pptx
3. Mokslinės publikacijos