TALLINNA TEHNIKAÜLIKOOL
INSENERITEADUSKOND
Virumaa kolledž

Manipulaator 4-DOF TRRR

Manipulator 4-DOF TRRR

MASINAEHITUS- JA ENERGIATEHNOLOOGIA PROTSESSIDE JUHTIMINE
LÕPUTÖÖ

Üliõpilane: Daria Molchanova

Üliõpilaskood: 207581EDJR

Juhendaja: Tatjana Baraškova, vanemlektor

Kohtla-Järve, 2024

AUTORIDEKLARATSIOON

Olen koostanud lõputöö iseseisvalt.

Lõputöö alusel ei ole varem kutse- või teaduskraadi või inseneridiplomit taotletud. Kõik töö koostamisel kasutatud teiste autorite tööd, olulised seisukohad, kirjandusallikatest ja mujalt pärinevad andmed on viidatud.

"27" jaanuar 2024.

Autor: Daria Molchanova

/ allkiri /

Töö vastab rakenduskõrgharidusõppe lõputööle/magistritööle esitatud nõuetele

"...." .... 20.........

Juhendaja: ....

/ allkiri /

Kaitsmisele lubatud

"...." .... 20.........

Kaitsmiskomisjoni esimees ....

/ nimi ja allkiri /

LIHTLITSENTS LÕPUTÖÖ ÜLDSUSELE KÄTTESAADAVAKS TEGEMISEKS JA REPRODUTSEERIMISEKS

Mina Daria Molchanova (sünnikuupäev: 05.08.1982)

1. Annan Tallinna Tehnikaülikoolile tasuta loa (lihtlitsentsi) enda loodud teose

Manipulaator 4-DOF TRRR mille juhendaja on Tatjana Baraškova,

1.1. reprodutseerimiseks säilitamise ja elektroonilise avaldamise eesmärgil, sealhulgas Tallinna Tehnikaülikooli raamatukogu digikogusse lisamise eesmärgil kuni autoriõiguse kehtivuse tähtaja lõppemiseni;

1.2. üldsusele kättesaadavaks tegemiseks Tallinna Tehnikaülikooli veebikeskkonna kaudu, sealhulgas Tallinna Tehnikaülikooli raamatukogu digikogu kaudu kuni autoriõiguse kehtivuse tähtaja lõppemiseni.

2. Olen teadlik, et punktis 1 nimetatud õigused jäävad alles ka autorile.

3. Kinnitan, et lihtlitsentsi andmisega ei rikuta kolmandate isikute intellektuaalomandi ega isikuandmete kaitse seadusest ja teistest õigusaktidest tulenevaid õigusi.

TalTech Inseneriteaduskond Virumaa kolledž

LÕPUTÖÖ ÜLESANNE

Üliõpilane: Daria Molchanova, 207581EDJR

Õppekava, peaeriala: EDJR 16/17 MASINAEHITUS- JA ENERGIATEHNOLOOGIA PROTSESSIDE JUHTIMINE

Juhendaja(d): Vanemlektor, Tatjana Baraškova, tatjana baraskova@taltech.ee

Lõputöö teema:

(eesti keeles) Manipulaator 4-DOF TRRR

(inglise keeles) Manipulator 4-DOF TRRR

Lõputöö põhieesmärgid:

1. Manipulaatori struktuuri analüüsi läbiviimine, otsese kinemaatika ülesande lahendamine mitmel viisil.
2. Digitaalsete võimaluste testimine ja kasutamine kinemaatika ülesande lahendamiseks ja simulatsiooni loomiseks.
3. Spetsiifilise manipulaatori mudeli alusel metoodiliste materjalide loomine inseneride õpetamiseks.

Lõputöö etapid ja ajakava:

Nr	Ülesande kirjeldus	Tähtaeg
1.	Lõputöö eesmärkide püstitamine	01.03.2024
2.	Manipulaatori struktuuri analüüsi tegemine ja vormistamine	25.03.2024
3.	Otsese kinemaatika ülesande lahendamine mitmel viisil, arvutuste tegemine ja vormistamine	22.04.2024
4.	Matlab'i koodi kirjutamine ja simulatsiooni loomine	29.04.2024
5.	Metaandmete esitamine	30.04.2024
4.	Graafiline osa, manipulaatori joonis, visuaaliseerimine	06.05.2024
5.	Lõputöö vormistamine	13.05.2024

Töö keel: eesti keel **Lõputöö esitamise tähtaeg:** "20"mai 2024a

Üliõpilane: Daria Molchanova .... "20"mai 2024a
/allkiri/

Juhendaja: Tatjana Baraškova .... “20”mai 2024a
/allkiri/

Programmijuht: Veroonika Shirokova .... “20”mai 2024a
/allkiri/

SISUKORD

EESSÕNA

Lõputöö teema pakkus välja vanemlektor Tatjana Baraškova. Lõputöö teema on seotud prantsuse ülikooliga «*Université de Picardie Jules Verne*» koostöös läbiviidava rahvusvahelise robootikaprojektiga.

Projekti eesmärkideks on erinevate manipulaatorite struktuuride analüüsimine ja kinemaatika ülesannete lahendamine. Seejärel otsustas autor selle projekti teema välja töötada ja ühe prantslastelt pakutud manipulaatori näitel ehitada matemaatilise mudeli, koostada programmi manipulaatori parameetrite arvutamiseks erinevate lähtetingimuste jaoks ning teostada ka manipulaatori töö simulatsiooni.

Autor tänab vanemlektori Tatjana Baraškovat nõuannete, Matlab'i programmiga seotud materjalide eest ning professor Didier Pascault lõputöö ülesannete lähteandmete eest, samuti loengutesarja ja teoreetiliste materjalide eest, mis aitasid selle projekti lõpule viia.

Märksõnad: manipulaator 4-DOF, TRRR, otsene kinemaatikaülesanne, Denavit-Hartenberg, Euleri nurgad, rotatsioonimaatriks, teisendusmaatriks, diplomitöö

LÜHENDITE JA TÄHISTE LOETELU

TRRR – (ing. Translation, Rotation, Rotation, Rotation) - manipulaator translatoorse, ja pöörde liigenditega.

DH – Denavit-Hartenberg

DH-parameetrid - Denavit-Hartenbergi parameetrid

θ, φ, ψ - Euleri pöördenurgad

x, y, z – Descartes'i koordinaadid

θ_i – Denavit-Hartenbergi parameeter, nurk z_i telje ümber vana x-telje suunas uuele x-teljele

d_i- nihke pikkus mööda z-telge ühisele normaalsele

a_i – tavanormaali pikkus

α_i – nurk tavanormaali ümber, vanast z-teljest uuele z-teljele

M_i^j- rotatsioonimaatriks

T_r - teisendusmaatriks

R – pöörlemine (ing. Rotation)

T – sirgjooneline liikumine (ing. Translation)

SISSEJUHATUS

Robotite kasutamine erinevates avaliku elu valdkondades kasvab pidevalt. Tööstuslikus tootmises on eriti oluline roll robotitel. Siin saab robotite kasutamisega oluliselt tõsta tootmise tootlikkust, samuti tagada toote valmistamise erinevate etappide täpsus ja ohutus. Roboteid on erinevat tüüpi: kortesi-, silindri-, SCARA-, polaar-, delta- ja liigendrobotid [1].

Liigendatud tööstusrobotid on manipulaatorid, millel on mitu liigendit, mida juhivad servomootorid. Teatud tüüpi selliste robotite liikumine jäljendab inimese kätt. Sellistel robotitel on mitmeid eeliseid, nagu suur kiirus, paindlikkus ja täpsus. Seda tüüpi roboteid kasutatakse peamiselt sellistes ülesannetes nagu monteerimine, keevitamine, värvimine ja toodete pakendamine [1].

Manipulaatori projekteerimisel valitakse esmalt manipulaatori kinemaatiline skeem ja teatud geomeetrilised parameetrid. Projekteerimisinseneril on vaja kindlaks määrata nihked, kiirused ja kiirendused kinemaatilistes paarides (kahes kõrvuti asetseva robotlüli liikuvat ühenduses) [2:12]. Selleks lahendatakse pöördkinemaatika ülesandeid. Ajamites nõutavad jõud aitavad määrata pöörddünaamika ülesande lahenduse. Edasi saab määrata lülide massikeskmete absoluutkiirused ja kiirenduse ning lülide nurkkiirused ja kiirenduse, lülide inertsjõudude momendid nende massikeskmetes. Saadud andmeid kasutatakse tugevusarvutuste tegemiseks. Nende ülesannete lahendamisel kasutatakse ligikaudseid geomeetrilisi parameetreid. Viimase sammuna peab projekteerimisinsener lahendama otsese ülesande, võttes arvesse valitud jõude. Kui sel juhul tehnilisi nõudeid ei täideta, kohandatakse manipulaatori konstruktsiooni [2: 17].

Roboti ehitamine koosneb ligikaudu järgmistest etappidest:

- tehniliste nõuete analüüs;
- manipulaatori kinemaatilise diagrammi ja geomeetriliste parameetrite valik;
- pöördülesannete lahendamine – ajamite ja juhtimissüsteemide nõuete määramine;
- ajamite projekteerimine;
- otsese ülesande lahendamine etteantud tingimustel;
- tehniliste nõuete täitmise kontroll;
- manipulaatori või juhtimissüsteemi konstruktsiooni muutmine, kui kontrolli tulemused ei ole rahuldavad [2: 17].

Käesolevas töös vaadeldakse üht projekteerimistöö etappi – otsese kinemaatika ülesande lahendamist.

Otsene kinemaatika ülesande lahendamine on näidetud nelja vabadusastmega (4-DOF) manipulaatori põhjal. Sellel manipulaatoril on neli liigendit, millest igaüks teeb teatud tüüpi liigutusi: sirgjoonelist liikumist või pöörlemist. Kõnealune manipulaator kannab nimetust TRRR, mis tähendab, et sellel on libisev liigend ja kolm pöörlevat liigendit [4, 5]. Seega on kõnealuste mehhanismide täielik nimetus TRRR 4-DOF manipulaator.

Seda tüüpi manipulaatorit kasutatakse tootmises ja sellel on mitmeid eeliseid, nagu suurem paindlikkus, võime teha erinevat tüüpi liigutusi, aga ka suhteliselt suur kandevõime [3]. Reeglina on manipulaatoritel RRR või RRRR struktuur. Sel juhul on kõik liigendid pöörlevad ja mehhanism jäljendab inimese kätt. Seda tüüpi roboteid toodavad paljud tootjad, sealhulgas FANUC, Mitsubishi, Yamaha ja muud (vt Joonis 1).

Joonis 1. Robotite näited (vasakult paremale: FANUC LR Mate 200iD [6], Mitsubishi RV-CR seria [7], Yamaha YAR3F [8])

TRRR manipilaator põhineb lineaarsel ühendusel, mis tagab manipulaatori liikumise piki horisontaaltelge. See lahendus võimaldab suurendada manipulaatori tööpiirkonda. Seda tüüpi manipulaatorit saab kasutada toorikute haaramiseks ja teisaldamiseks ühest tööpiirkonna osast teise, millele järgneb teisaldatud tooriku paigaldamine. Viimase lüli keeramine võimaldab pööratavat toimingut. Sellise roboti ligikaudne konstruktsioon on näidatud joonisel 2.

Joonis 2. Yamaha robot [9]

Lõputöö eesmärk on TRRR-manipulaatori struktuuri analüüs ja matemaatilise mudeli konstrueerimine. Käesolev töö on õppematerjal, mida saab kasutada robootika aluste õpetamise osana. Töös esitatud metoodiline materjal aitab paremini mõista puustruktuuriga tüüpi manipulaatorite konstruktsiooni iseärasusi ja aitab kaasa tüüpiliste projekteerimisprobleemide lahendamisele.

Töös autor esitas ja lahendas kinemaatilisi ülesandeid ning näitas nende lahendamise algoritmi. Aluseks võeti prantsuse ülikooli poolt antud manipulaatori mõõtmed ja struktuur. Seda konkreetset etteantud mõõtmetega manipulaatorit näitena kasutades autor lahendas samm-sammult kinemaatilise ülesande, koostas programmikoodi ja virtuaalsed simulatsioonid. Seda algoritmi saab kasutada nii metoodiliste materjalidena õppimisel kui ka päris roboti kujundamisel. Erinevad keevitus- ja tõste- ning transpordiseadmed on sama puustruktuuriga manipilaatorid ja nende struktuuri mõistmine aitab tulevastel konstruktoriinseneritel seda ametit omandada.

1. MANIPULAATORI STRUKTUUR

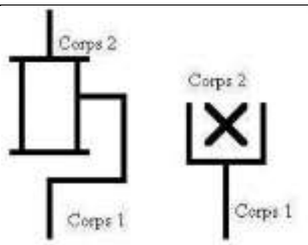
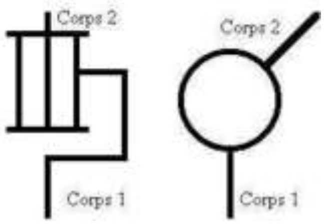
Manipulaator on mehaaniline süsteem, mis on loodud objektide manipuleerimiseks. Manipulaatorid jäljendavad inimese, näiteks inimese käe või peopesa liikumist. Võimalus reprodutseerida inimese toiminguid saavutatakse manipulaatorile mitme vabadusastme andmisega. Üks kuulsamaid manipulaatoreid on robotkäsi [10: 7].

Robotkäsi (vt Joonis 1) on manipulaator, mis koosneb alusest, liigenditest, lülidest ja haaratsist. Alus on manipulaatori osa, mis ühendab selle kinnituspinnaga. See võib olla fikseeritud või mitte. Lingid – roboti käe komponendid on kindla pikkusega ja on omavahel ühendatud liigeste abil. Kahte külgnevat lüli ühendavad liigendid teostavad käe liikumist. Viimane lüli toetab manipulaatori haaratsit. Haarats on konstruktsiooni viimane element. Seda kasutatakse manipulaatori teatud tööriista hoidmiseks [11].

Manipulaator on avatud kinemaatiline ahel, kus iga lüliga, välja arvatud esimene ja viimane, on kaasas eelkäija ja järglane. Esimesel lingil on ainult järglane ja viimasel lingil ainult eelkäija [12: 9]. Kinemaatiline ahel kirjeldab manipulaatori geomeetriat. See on manipulaatori linkide jada graafiline esitus, mis on omavahel ühendatud liigenditega (vt Joonis 3).

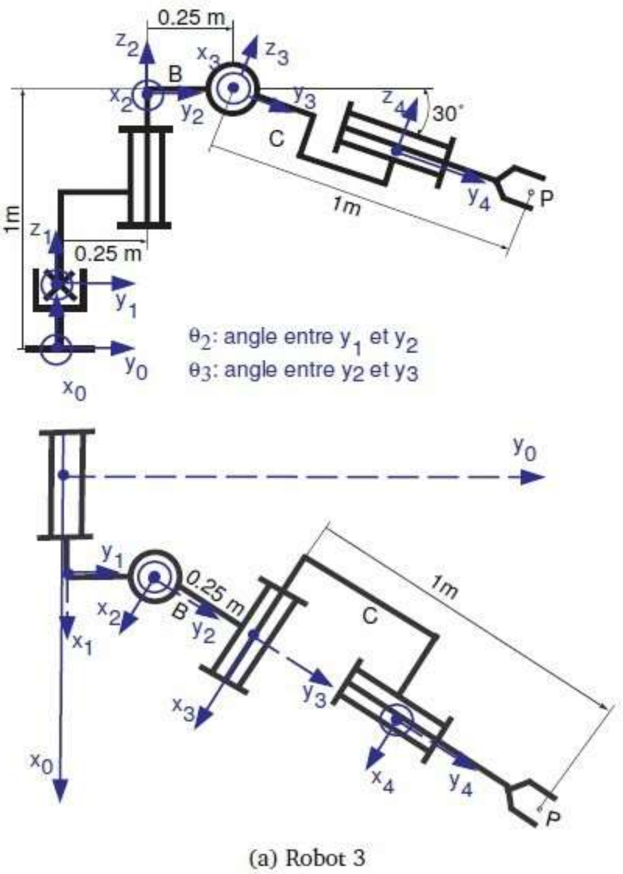
Liigendid võivad olla pöörlevad või sirgjoonelised, esimesel juhul nimetatakse neid *revolute joints*, teisel juhul *prismatic joints* [13: 9]. Pöördliigendi puhul määratakse külgnevate lülide suhteline asend nurgamuutuja, sirgjoonelise liigendi puhul lineaarse nihkega. Mõlemal juhul nimetatakse neid muutujaid üldistatud koordinaadid. Kõikide manipulaatori üldistatud koordinaatide kogumit, mis seda ruumis ainulaadselt määratlevad, nimetatakse manipulaatori konfiguratsiooniks [13: 9]. Tabelis 1.1. on esitatud sirgjooneliste ja pöörlevate liigendite tähistamine.

Tabel 1.1. Liigendite tähistamine [14]

Liigendi sümbol	Tähendus	Liigendi sümbol	Tähendus
	Sirgjooneline (translaatorne, libisev) liigend *Prismatic joint*		Pöörlev liigend *Revolute joint*

1.1 Lähteandmete analüüs

See töö põhineb konkreetsel manipulaatoril. Algtingimused ja mõõtmed oli antud Prantsuse ülikool «*Université de Picardie Jules Verne*» poolt graafilisel kujul (vt Joonis 1.1).



Joonis 1.1 Manipulaatori kinematiline skeem [14]

Käsitletaval manipulaatoril on 4 vabadusastet, mis võimaldab teostada erinevaid toiminguid kolmemõõtmelises ruumis. Manipulaator koosneb 4 liigendist. Manipulaatori esimene kinemaatiline paar tagab sirgjoonelist ja kolm järgmist pöörlevat liikumist. Manipulaatori väljundlüliks on tööriista haarats.

Kõrvuti asetsevad lülid on ühendatud liikuvate liigenditega - kinemaatiliste paaridega. Vaadeldavas seadmes on kinemaatiliste paaride arv 4. Selle mehhanismi kinemaatilised paarid on üksikult liikuvad või viienda klassi paarid [2: 12]. Need võimaldavad suhtelist pöörlevat või sirgjoonelist liikumist. Vastavalt ISO 10360-12 ja ISO 8373:2012 standardile on liigendid tähistatud *T-Transition* - sirgjoonelist liikumist ja *R-Rotation* - pöörlemist teostavad kinematilised paarid [4, 5]. Lähtendmete põhjal on antud töös käsitletav manipulaator tähistatud kui TRRR 4-DOF manipulaatorit.

Seadmel on puustruktuur. See on hierarhiline struktuur, kus iga eelnev sõlm annab tingimuse järgmise sõlme liikumiseks. Sarnase ehitusega on ka teised tööstuslikud liigendrobotid [2: 14].

1.2 Ülesande püstitamine

Uurimise ülesande püstitas prantslane professor Didier Pascault koostöö projekti raames.

Ülesanne oli kirjutada üles geomeetriline mudel, kasutades erinevaid meetodeid - leida positsioonivõrrandid x, y ja z koordinaatides ning tööriista orientatsiooni (kolm Euleri nurka x-, y ja z-telje ümber) [14].

Lähtudes määratud ülesannetest autor uuritas lõputöö raames konkreetse manipulaatori struktuuri ja katsutas mitu meetodit otsese kinemaatika ülesande lahendamiseks.

Lõputöö käigus käsitles autor järgmisi otsese kinemaatika ülesande lahendamise maatriksmeetodeid:

- Rotatsioonimaatriksi meetodit
- Denavit-Hartenbergi meetodit

Autor teostas arvutused ka Euleri nurkade jaoks (määramine ära haaratsi asukoha toe suhtes).

Saadud tulemused autor esitas matemaatiliste arvutuste, programmikoodi ja virtuaalse simulatsiooni kujul. Arvutused tehti käsitsi, Excel'i ja Matlab'i programmi abil. Selles töös ei käsitlenud autor roboti võimsuse arvutamist, selliste näitajate määramist nagu kandevõime ja muud.

2 MANIPULAATORI KINEMAATIKA ÜLESANDED

Manipulaatori kinemaatika kirjeldab manipulaatori liikumist rakendatud jõude arvestamata. Sel juhul võetakse arvesse lülide asukohti, kiirusi, kiirendusi ja nende tuletisi [12:30].

Manipulaatori asendi kinemaatilisel analüüsimisel ja projekteerimisel on vaja lahendada kaks ülesannet: otsest kinemaatika ülesannet ja pöördkinemaatika ülesannet [12: 30].

Pöördkinemaatika ülesanne (vt Joonis 2.1) hõlmab pöördteisendusvõrrandi lahendamist, et leida seoseid manipulaatori lülide vahel sõltuvalt käe asendist ruumis. Pöördkinemaatikas on antud iga lüli pikkus ja punkti asukoht töömahus (tavaliselt haaratsi või töövahendiga seotud lõpliku koordinaatide süsteemi asukoht) ning tuleb arvutada iga manipulaatori liigendi üldistatud koordinaatide komplekt [11, 12: 30, 13:10].

Joonis 2.1 Pöördkinemaatika ülesande skeem

Otsene kinemaatika ülesanne (vt Joonis 2.2) hõlmab otsese teisendusvõrrandi lahendamist, et määrata linkide absoluutsed asukohad nurkade ja linkide vaheliste nihete järgi ehk linkide positsioonid fikseeritud koordinaatsüsteemis. Otsese ülesande algtingimusteks on teatud ajahetkel teadaolevad üldistatud koordinaatide väärtused, mis määravad linkide asukoha üksteise suhtes. Lingide geomeetrilised mõõtmed ja liigendite nurgad loetakse teadaolevaks. On vaja kindlaks määrata manipulaatori viimase lingi (haaratsi või töövahendiga seotud koordinaatide süsteemi) asukoha ja orientatsiooni [11, 12: 30, 13: 10, 15: 5].

Joonis 2.2 Otsese kinemaatika ülesande skeem

Otsese kinemaatika ülesande lahendamiseks on mitu meetodit:

- Geomeetriline meetod
- Maatrikimeetodid
    - Rotatsioonimaatriksi meetod
    - Denavit-Hartenbergi meetod

2.1 Rotatsioonimatriksi meetod (otsene meetod)

2.1.1 Meetodi kirjeldus

Iga liigendile kinnitame koordinaadistiku, mis pöörleb või liigub koos liigendiga. Liigendi liikuvus toimub paremale suunatud koordinaatsüsteemis [14].

Manipulaatori geomeetriline mudel põhineb geomeetriliste muutujate ja liigendite muutujate vahelisel seosel [14].

Liigendite muutujad on

- pöördliikumised. Tähistatud pöördenurgaga (siin töös on θ)
- kulgliikumised. Tähistatud distantsiga (siin töös on T_r) [14]

Geomeetrilised muutujad kirjeldavad manipulaatori lõpplüli – haaratsi asukohta ja orientatsiooni [14].

Manipulaatori geomeetriline mudel määrab tööriista teljestiku asukoha selle baasteljestiku (R_b) suhtes. Baasteljestik on ühendatud manipulaatori alusega. Geomeetriline mudel määrab tööriista asukoht manipulaatori tööruumis ja annab üldiselt manipulaatori geomeetrilise struktuuri esituse [12: 24, 14].

Haaratsi asukoha saab esitada funktsiooniga

$$\vec{r} = f(\overrightarrow{\theta(t)}), \tag{2.1}$$

kus

$$\vec{\theta} = \begin{bmatrix} \theta_1(t) \\ \theta_2(t) \\ \dots \\ \theta_n(t) \end{bmatrix} \qquad \vec{r} = \begin{bmatrix} r_1 \\ r_2 \\ r_3 \end{bmatrix}$$

Vektor $\vec{\theta}$ on liigendi muutujate vektor ja vektor $\vec{r}$ annab manipulaatori lõppefektori (haaratsi) asukoha, koordinadistikus $\vec{r} = (x, y, z)$ [14].

Manipulaatori lõppefektori orientatsioon baaskoordinadistikus $R_0(x_0 y_0 z_0)$ saab esitada funktsiooniga:

$$\vec{\alpha} = f(\overrightarrow{\theta(t)}), \tag{2.2}$$

kus

$$\vec{\theta} = \begin{bmatrix} \theta_1(t) \\ \theta_2(t) \\ \dots \\ \theta_n(t) \end{bmatrix} \qquad \vec{\alpha} = \begin{bmatrix} \varphi \\ \theta \\ \psi \end{bmatrix}$$

2.1.2 Otsese meetodi rakendamine

Igale liigendile kinnitame raami R_i. Koordinaatide telgede suund peab olema kogu struktuuri jaoks sama: parem- või vasakukäeline [14].

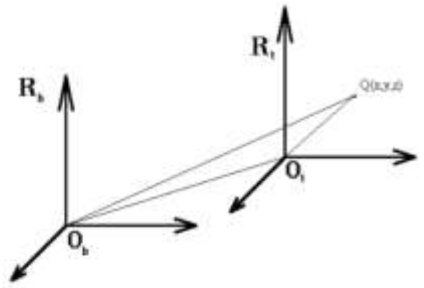
Sel juhul tuleks võimaluse korral järgida järgmisi soovitusi:

Kui see on võimalik:

- libisevates liigendites koordinaattelgede asukoht on paralleelne eelmise teljestikuga.
- pöördliigendites pöörlemistelg on sama nimega kui paralleelne telg eelmises teljestikus [14].

Kinnitame alusele liigendile koordinaadistiku R_b ja lõppefektorile (haaratsile) koordinaadistiku R_t. Telgede asukoha paremaks nägemiseks tuleks manipulaatori kinemaatilist diagrammi kujutada kahes vaates, näiteks eest- ja pealtvaates.

Tähistame O_b-baaskoordinaadistiku keskpunkt, O_t – viimase koordinaadistiku keskpunkt, Q – manipulaatori tööriista positsioon (vt joonis 2.3) [14].



Joonis 2.3 Tööriista posistsioon avaldatud vektorina [14]

Seejärel kirjutame asukohavektori välja naaberkoordinaadisüsteeme ühendavate vektorite summana:

$$\overrightarrow{O_b Q}_{/R_b} = \overrightarrow{O_b O_1}_{/R_b} + \overrightarrow{O_1 O_2}_{/R_b} + \overrightarrow{O_2 O_3}_{/R_b} + \cdots + \overrightarrow{O_\iota O_{\iota+1}}_{/R_b} + \cdots + \overrightarrow{O_t \mathrm{Q}}_{/R_b} \quad (2.3)$$

Avaldame vektorid $\overrightarrow{O_\iota O_{\iota+1}}_{/R_b}$ koordinaadistikustes R_i, kasutades rotatsioonmatrikseid [14].

2.1.3 Rotatsioonimaatriksid

Iga manipulaatori punkti koordinaadid saab kirjeldada vektoriga $\vec{P}$, millega tähistatakse projektsioonid baaskoordinadistiku x, y ja z telgedel [12: 23].

$$\vec{P}^A = \begin{bmatrix} p_x \\ p_y \\ p_z \end{bmatrix} \quad (2.4)$$

Kolm ühikvektorit määravad keha orientatsiooni ristkoordinaadistikus {A} (2.5). Need vektorid moodustavad kehaga seotud koordinastiku {B}. Ühikvektoritega saab kirjedada keha orientatsiooni muutmine, kui üks koordinadistik pöörab teise koordinaadistiku suhtes. Iga ühikvektor saab esitada koordinaadistikus {A} tema projektsiooniga x, y, z-telgedel [12:23].

$$\vec{X}_B^A = \begin{bmatrix} x_B^x \\ x_B^y \\ x_B^z \end{bmatrix} \qquad \vec{Y}_B^A = \begin{bmatrix} y_B^x \\ y_B^y \\ y_B^z \end{bmatrix} \qquad \vec{Z}_B^A = \begin{bmatrix} z_B^x \\ z_B^y \\ z_B^z \end{bmatrix} \tag{2.5}$$

Kogu need kolm vektorit moodustavad rotatsioonimaatriksi, mis näitab koordinadistiku {B} orientatsioon koordinaadistiku {A} suhtes [12: 23].

$$M_B^A = \begin{bmatrix} x_B^x & y_B^x & z_B^x \\ x_B^y & y_B^y & z_B^y \\ x_B^z & y_B^z & z_B^z \end{bmatrix} \tag{2.6}$$

Koordinaadistiku {B} asend teljestiku {A} suhtes saab kirjeldada selle teljestiku alguspunkti asendvektoriga $\vec{P}_{BO}^A$ ning rotatsioonimaatriksiga M_B^A [12: 24].

$$\{B\} = \{\, M_B^A, \vec{P}_{BO}^A\} \tag{2.7}$$

Asendvektorit $\vec{P}^B$ saab esitada koordinaadistikus {A} kahe vektori summana [12: 24].

$$\vec{P}^A = \vec{P}^B + \vec{P}_{BO}^A \tag{2.8}$$

Kui teljestik {B} on pööratud, siis asendvektorit $\vec{P}^B$ saab esitada koordinaadistiku {A} rotatsioonimaatriksi M_B^A abil. Vektor $\vec{P}^A$ on rotatsioonimaatriksi M_B^A ja asendivektori $\vec{P}^B$ korrutis. [12:24].

$$\vec{P}^A = M_B^A \cdot \vec{P}^B \tag{2.9}$$

Rotatsioonimaatriks ümber x-telje

$$M(x,\theta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\theta & -sin\theta \\ 0 & sin\theta & cos\theta \end{bmatrix}, \tag{2.10}$$

kus

θ - pöördenurk kahe koordinaadistiku vahel ümber x-telje pööramisel [12: 23].

Rotatsioonimaatriks ümber y-telje

$$M(y,\psi) = \begin{bmatrix} cos\psi & 0 & sin\psi \\ 0 & 1 & 0 \\ -sin\psi & 0 & cos\psi \end{bmatrix}, \tag{2.11}$$

kus

ψ - pöördenurk kahe koordinaadistiku vahel ümber y-telje pööramisel [12: 24].

Rotatsioonimaatriks ümber z-telje

$$M(z,\varphi) = \begin{bmatrix} cos\varphi & -sin\varphi & 0 \\ sin\varphi & cos\varphi & 0 \\ 0 & 0 & 1 \end{bmatrix}, \tag{2.12}$$

kus

φ - pöördenurk kahe koordinaadistiku vahel ümber z-telje pööramisel [12: 24].

Kui pööratatakse R_0 teljestiku ümber telje niimoodi et see muutub R_1 siis rotatsioonimaatriks saab kirjutada niimoodi: [14].

$$M_0^1: R_0 \to R_1 \tag{2.13}$$

Mitme järjestikuse pöörde korral kirjutatakse maatriks: [14].

$$M_i^{i+\lambda} = M_i^{i+1} M_{i+1}^{i+2} \dots M_{i+\lambda-1}^{i+\lambda} \tag{2.14}$$

2.1.4 Põhivalem

Punkti koordinaadid leitakse vektorite summana

$$\overrightarrow{O_b Q}_{/R_b} = \overrightarrow{O_b O_t}_{/R_b} + M_b^t \overrightarrow{O_t Q}_{/R_t}, \tag{2.15}$$

kus

$$M_b^t \overrightarrow{O_t Q}_{/R_t} = \overrightarrow{O_t Q}_{/R_b}$$

Teades rotatsioonimaatrikse, kirjutame välja valemi mehaanilise konstruktsiooni geomeetrilise mudeli arvutamiseks [14].

$$\overrightarrow{O_b Q}_{/R_b} = \overrightarrow{O_b O_1}_{/R_b} + M_b^1 \overrightarrow{O_1 O_2}_{/R_1} + M_b^2 \overrightarrow{O_2 O_3}_{/R_2} + \dots + M_b^i \overrightarrow{O_\iota O_{\iota+1}}_{/R_i} + \dots + M_b^t \overrightarrow{O_t \mathrm{Q}}_{/R_t} \tag{2.16}$$

2.2 Denavit-Hartenbergi meetod

Denavit-Hartenbergi meetodi põhiidee seisneb selles, et otsese kinemaatika ülesande saab lahendada, teades suhtelisi nihkeid iga lingi külge kinnitatud koordinaatsüsteemide vahel [15:585].

Denavit-Hartenbergi meetodi olemus on määrata kindlaks muutujad, mis määravad üheselt manipulaatori arhitektuuri ja selle lülide asukoha [15:585].

Otsese kinemaatika saab lahendada valemiga:

$$T_{0n}(\theta_1 \dots \theta_n) = T_{01}(\theta_1) T_{12}(\theta_2) \cdots T_{n-1,n}(\theta_n), \tag{2.17}$$

kus

$T_{i,i-1} \in SE$ tähistab suhtelisi liikumisi koordinaadistiku {i-1} ja {i} vahel [15: 585].

Iga koordinaadistiku teisendamine järgmiseks viiakse läbi vastavalt valemile ja seda kirjeldatakse mitme parameetriga, nii öeldes Denavit-Hartenbergi parameetritega.

Denavit-Hartenbergi meetod seisneb:

- lülide koordinaattelgede määramises teatud viisil;
- kahe parameetri komplekti määramises (üks parameetri komplekt kirjeldab linkide asukohta, teine komplekt määrab nende orientatsiooni);
- kahe parameetri komplekti vahelise seose määramises oma geomeetrilise mudeli kaudu [15: 585].

2.2.1 Koordinaattelgede määramine

Lülid on nummerdatud järjekorras: 0,1,...,n. Lüli 0 on fikseeritud alus ja lüli n on lõppefektor. Saime, et manipulaator koosneb n+1 lülist, mis on omavahel ühendatud n-kinemaatilise paariga.

Iga liigendi jaoks määrame koordinaatide süsteemi vastavalt järgmistele reeglitele:

- z_i-telg seotud i-liigendiga ja on suunatud piki pöörleva liigendi pöörlemistelge või mööda libiseva liigendi liikumissuunda;
- x_i-telg on ühine risti vastavalt z_i ja z_{i-1} teljega;
- x_i-telg lõikub z_i ja z_{i-1} telgedega;
- telgede y suund moodustab x-teljega paremale suunatud teljestikut.

Koordinaatide telgede asukoha määramisel võib tekkida olukordi, kus vastastikku risti olev joon ei ole defineeritud või ei ole ainus [15: 587].

Erijuhtumid:

1. Kui külgnevate pöörlevate liigendite teljed ristuvad

Kui pöördliigendi kaks kõrvuti asetsevat telge ristuvad, siis liigendi telgede vahel vastastikku risti asetsevat joont ei eksisteeri. Sel juhul võtame lingi pikkuse, mis on võrdne nulliga ja valime x_i nii, et see oleks risti tasapinnaga, mida katavad teljed z_i ja z_{i-1}. [15: 587].

2. Kui külgnevate pöörlevate liigendite teljed on paralleelsed

Teine erijuhtum tekib siis, kui pöörleva liigendi kaks kõrvuti asetsevat telge on paralleelsed. Sel juhul valitatakse risti asetsev joon, mis annab suurima arvu parameetreid, mis võrdub nulliga (mis lihtsustab edasisi arvutusi) [15: 587].

3. **Prismaatilised ühendused**

   Prismaatiliste ühenduste jaoks valitakse lingi z-suund mööda positiivset liikumissuunda. Selle valiku korral on ühenduse taane d_i muutuv ja nurk θ_i on konstant [15: 588].

4. **Baas- ja sihtkoordinaatide telgede määramine**

   Kindlad reeglid selle kohta, kuhu manipulaatori aluse ja otsalüli koordinaatteljed suunata puuduvad. Sel juhul tuleks valitatakse niisugune suund millega suurim Denavit-Hartenbergi parameetrite arv võrdub nulliga[15:588].

2.2.2 Denavit-Hartenberg parameeters

Iga liigend sisaldab kolme konstantset parameetrit (θ_i, a_i, α_i), ja ühte muutujat d_i [15: 591].

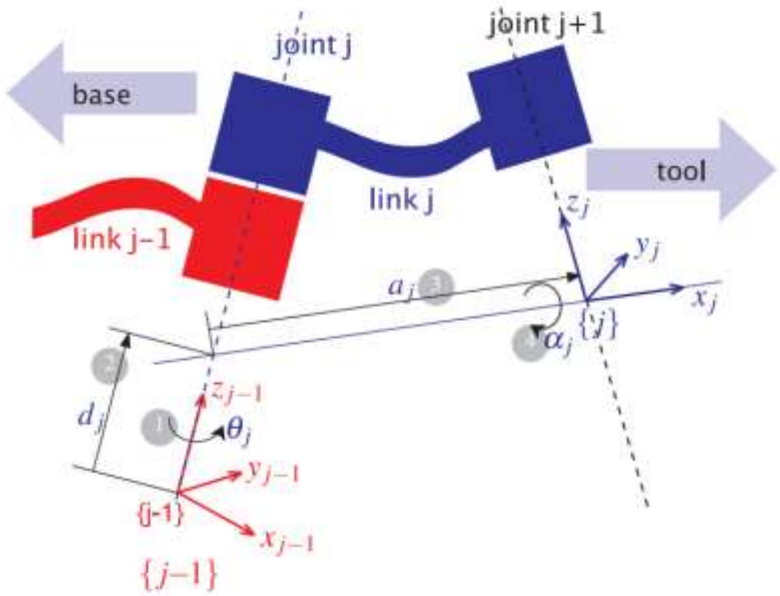
Parameetrite märkimiseks peab vaatlema naaberkoordinaadistike. Koordinaatide referentssüsteem {i-1} teisendatakse koordinaatide referentssüsteemiks ja järgmiste teisendustega:

- uue koordinaadistiku ülekandmine piki selle z-telge kaugusele d_i;
- uue koordinaadistiku pööramine ümber oma z-telje nurga θ_i võrra;
- uue koordinaadistiku ülekandmine piki selle x-telge kauguse a_{i-1} võrra;
- koordinaadistiku {i-1} pööramine ümber oma x-telje nurga α_{i-1} võrra [15: 591].

Teisendamised moodustavad Denavit-Hartenbergi (DH) parameetrid (vt Joonis 2.4).

- d_i- on kaugus piki z_{i-1} telge i-1 ja i-teljestiku keskpunktide vahel;
- θ_i – on pöördenurk z_{i-1} telgede ümber x_{i-1} ja x_i vahel;
- a_i – on kaugus piki x_i -telge z_{i-1} ja zi telgede vahel (normaali pikkus);
- α_i – on pöördenurk ümber x_i telgede z_{i-1} ja z_i vahel [15: 591].

Avatud ahela puhul, millel on neli ühe vabadusastme liigendit, on neli DH parameetrid piisavad, et täpselt kirjeldada kinemaatikat [15: 591].



Joonis 2.4 Denavit-Hartenberg parameetrid [16: 219]

Denavit-Hartenbergi parameetrid sisestatakse tabelisse. Neljast lingist koosneva manipulaatori puhul koosneb tabel 5 reast – vastavalt koordinaatsüsteemide arvule. T_0 süsteem on ühendatud manipulaatori alusega ja T_5 koordinadistiku on ühendatud tööriista käepidemega.

Tabel 2. Denaviit-Hartenbergi parameetrid. Üldine struktuur

i ($T_{i-1} \rightarrow T_i$)	θ_i	d_i	a_i	α_i
i_1 ($T_0 \rightarrow T_1$)	θ_1	d_1	a_1	α_1
i_2 ($T_1 \rightarrow T_2$)	θ_2	d_2	a_2	α_2
i_3 ($T_2 \rightarrow T_3$)	θ_3	d_3	a_3	α_3
i_4 ($T_3 \rightarrow T_4$)	θ_4	d_4	a_4	α_4
I_5 ($R_4 \rightarrow R_5$)	θ_5	d_5	a_5	α_5

2.2.3 Teisendusmaatriksid

Seejärel kasutati D-H parameetreid homogeensete teisendusmaatriksite (T_1^0, T_2^1, T_3^2, T_4^3, T_5^4) arvutamiseks asendades üldvalemis T_i^{i-1} [15: 593].

Algebraliselt kirjutatud teisendused näevad välja järgmiselt.

$$T_{i-1,i} = Rot(\hat{x}, \alpha_{i-1}) Trans(\hat{x}, a_{i-1}) Trans(\hat{z}, d_i) Rot(\hat{z}, \theta_i) =$$

$$= \begin{bmatrix} cos\theta_i & -sin\theta_i cos\alpha_i & sin\theta_i sin\alpha_i & a_i cos\theta_i \\ sin\theta_i & cos\theta_i cos\alpha_i & -cos\theta_i sin\alpha_i & a_i sin\theta_i \\ 0 & sin\alpha_i & cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.18)$$

$$\text{Siin} Rot(\hat{x}, \alpha_{i-1}) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & cos\alpha_{i-1} & -sin\alpha_{i-1} & 0 \\ 0 & sin\alpha_{i-1} & cos\alpha_{i-1} & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.19)$$

$$Trans(\hat{x}, a_{i-1}) = \begin{bmatrix} 1 & 0 & 0 & a_{i-1} \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.20)$$

$$Trans(\hat{z}, d_i) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.21)$$

$$Rot(\hat{z}, \theta_i) == \begin{bmatrix} cos\theta_{i-1} & -sin\theta_{i-1} & 0 & 0 \\ sin\theta_{i-1} & cos\theta_{i-1} & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.22)$$

Korrutades homogeensed teisendusmaatriksid õiges järjekorras T_5^0 saab saada lõppefektori koordinaadistiku [11, 14].

$$T_5^0 = T_1^0 T_2^1 T_3^2 T_4^3 T_5^4 \quad (2.23)$$

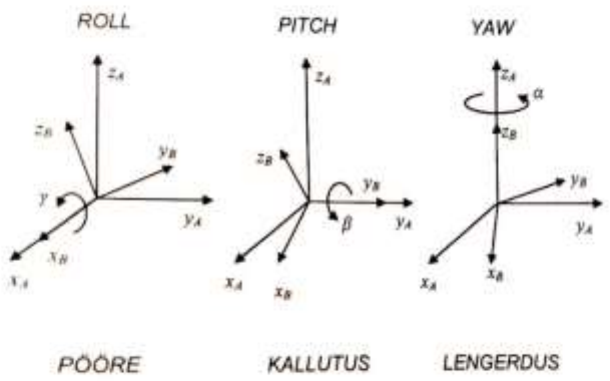
Maatriksite järjestikuse korrutamisel saame manipulaatori lõpliku maatriksi

$$T_5^0 = \begin{bmatrix} R(\theta) & P(\theta) \\ 0 & 1 \end{bmatrix}, \quad (2.24)$$

kus $P(\theta)$ esitab lõppefektori positsiooni, $R(\theta)$ esitab lõppefektori rotatsiooni [11, 14].

2.3 Euleri nurgad

Euleri teoreemi kohaselt võib suvalist pöörlemist kujutada mitte rohkem kui kolme pöördena ümber erinevate koordinaattelgede. Sellest järeldub, et teljestiku suvalise pöörde saab jaotada kolmeks sõltumatudeks pöördenurgaks [12: 28]: pöörde- (*roll*), kallutus- (*pitch*), lengerdusnurk (*yaw*). Pöördenurk asub yz-tasandil ja mõõdetakse teljestiku pööramisel x-telje ümber, kallutusnurk asub xz-tasandil ja mõõdetakse teljestiku pööramisel y-telje ümber, lengerdusnurk asub xy-tasandil ja mõõdetakse teljestiku pööramisel z-telje ümber (vt Joonis 2.5). Vastavalt pöörlemisteljele nimetatakse nurki α, β, γ või x-, y-, z- nurkadeks [12: 28].



Joonis 2.5 Eyleri nurgad [12: 28]

$$R(\alpha,\beta,\gamma) = IRot(\hat{z},\alpha)Rot(\hat{y},\beta)Rot(\hat{x},\gamma), \tag{2.25}$$

kus

$$Rot(\hat{z},\alpha) = \begin{bmatrix} cos\alpha & -sin\alpha & 0 \\ sin\alpha & cos\alpha & 0 \\ 0 & 0 & 1 \end{bmatrix}, \quad Rot(\hat{y},\beta) = \begin{bmatrix} cos\beta & 0 & sin\beta \\ 0 & 1 & 0 \\ -sin\beta & 0 & cos\beta \end{bmatrix},$$

$$Rot(\hat{x},\beta) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\gamma & -sin\gamma \\ 0 & sin\gamma & cos\gamma \end{bmatrix}$$

$$Rot(\alpha,\beta,\gamma) = \begin{bmatrix} c_\alpha c_\beta & c_\alpha s_\beta s_\gamma - s_\alpha c_\gamma & c_\alpha s_\beta c_\gamma + s_\alpha s_\gamma \\ s_\alpha c_\beta & s_\alpha s_\beta s_\gamma + c_\alpha c_\gamma & s_\alpha s_\beta c_\gamma - c_\alpha s_\gamma \\ -s_\beta & c_\beta s_\gamma & c_\beta c_\gamma \end{bmatrix}$$

$$r_{11}^2 + r_{21}^2 = cos^2\beta,\ cos\beta \neq 0,\ \beta \neq \pm 90^\circ \tag{2.26}$$

$\beta = atan2(-r_{31}, \sqrt{r_{11}^2 + r_{21}^2})$ and $\beta = atan2(-r_{31}, -\sqrt{r_{11}^2 + r_{21}^2})$

Esimesel juhul on nurk β vahemikus $[-90°, 90°]$, teisel juhul on vahemikus $[90°, 270°]$.

Eeldusel, et $r_{31} \neq \pm 1$, nurk $\beta \neq \pm 90°$ ja on vahemikus $[-90°, 90°]$, β saab leida valemist

$$\beta = atan2(-r_{31}, \sqrt{r_{11}^2 + r_{21}^2})$$

α ja γ on leitav vastavatest valemitest

$$\alpha = atan2(r_{21}, r_{11})$$

$$\gamma = atan2(r_{32}, r_{33})$$

Juhul et $\beta = \pm 90°$ eksisteerib α ja γ jaoks on ainult üks lahenduse tüüp [12: 28].

Juhul kui $r_{31} = -1$, $\beta = 90°$ ainuke võimalik lahendus et $\alpha = 0$ ja $\gamma = atan2(r_{12}, r_{22})$. Juhul kui $r_{31} = 1$, $\beta = -90°$ ainuke võimalik lahendus et $\alpha = 0$ ja $\gamma = -atan2(r_{12}, r_{22})$ [15:577].

3 KINEMAATIKA OTSESE ÜLESANDE LAHENDAMINE

3.1 Rotatsioonimaatriksi meetodi kasutamine

Ülesanne kujutab endast liigendikäe manipulaatorit, mis on järjestikku ühendatud liigendite kett, millest igaühel on üks vabadusaste.

3.1.1 Telejstiku määramine

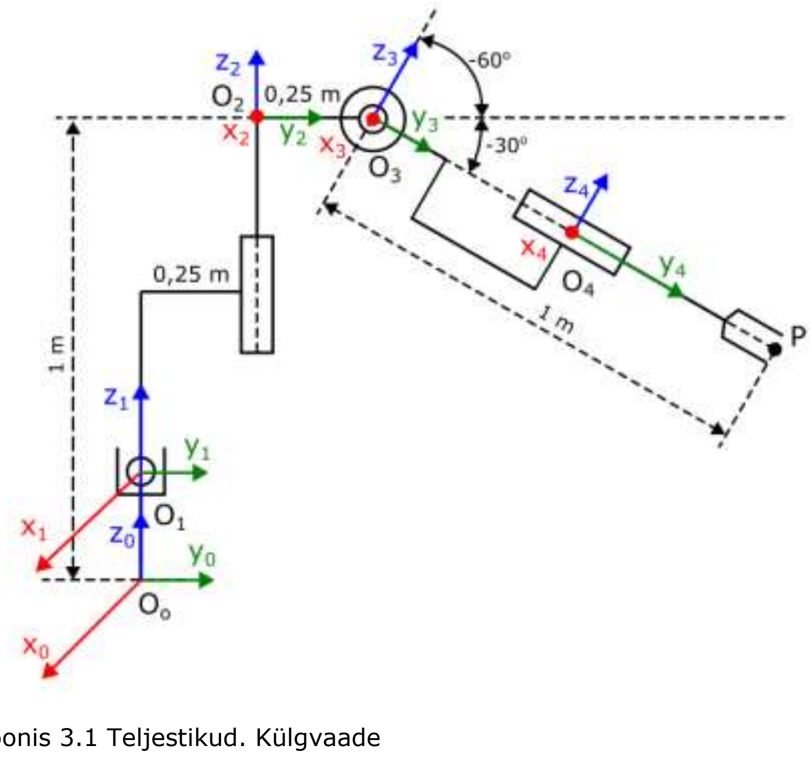
Tähistame P-tähega manipulaatori töötööriista asukoha. Määrame igale liikuvale lülile koordinaadistiku R_i.

Püüame võimalusel järgida järgmisi reegleid:

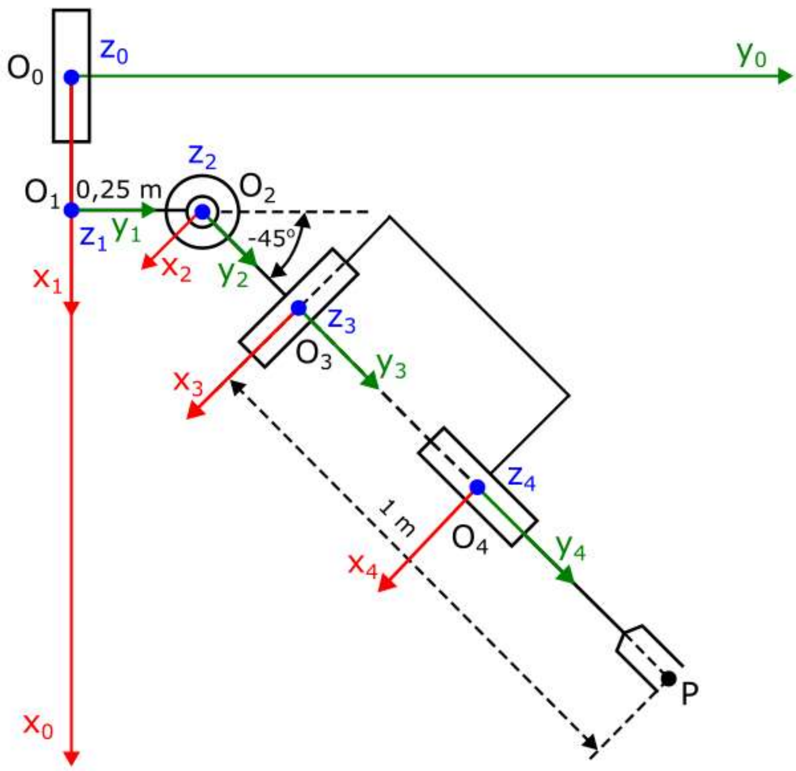
1. sirgjoonelises liigendis teljestik on liikumissuunaga paralleelne;
2. pöördliigendis kannab koordinadistiku pöörlemistelg sama nime kui eelmises koordinadistikus selle teljega paralleelne telg;
3. koordinaadistiku orientatsioon on kõikjal ühesugune.

Teljestiku suunda oli pakkutud lähteandmetes. Telgede asukohas vastuolusid ei leitud, seega kasutas autor sarnast paigutust ka ülesande lahendamise esimeses meetodis. Edasiste toimingute jaoks on mugav ja sageli vajalik omada kahte manipulaatori vaadet. Sel juhul kasutas autor ka kahte vaadet: külg- ja pealtvaade.

Telgede suund on näidatud joonistel 3.1, 3.2.



Joonis 3.1 Teljestikud. Külgvaade



Joonis 3.2 Teljestikud. Pealtvaade

Baaskoordinaadistik R_0 on antud. See on Descartes'i, paremale orienteeritud süsteem.

Koordinaadistik R_1 ühendatud esimese sirgjoonelise liigendiga. Selle teljed on suunatud paralleelselt baaskoordinaadistikuga. Teise liigendi telg z_2 on suunatud piki selle pöörlemistelge ja see on kollineaarne teljega z_1. Kolmanda liigendi x_3-telg on suunatud piki pöörlemistelge ja paralleelselt teise liigendi x_2-teljega. R_4 koordinaadistik ei muutu orientatsiooni R_3 suhtes. Neljanda liigendi teljestik on paralleelne R_3-ga. y_3 ja y_4 teljed on kollineaarsed.

Esimene samm on otsustada millise telje ümber tuleb pöörata koordinaadistiku R_i nii, et see on samuti orienteeritud nagu koordinaadistik R_{i+1}. θ on pöördenurk koordinaadistiku R_i et see saab orienteeruda samamoodi kui R_{i+1}.

1. Liigend 1 liigub piki x-telge $\theta_1 = 0$
2. Liigend 2 pöörleb ümber z-telje $\theta_2(y1, y2) - negativ = -45°$
3. Liigend 3 pöörleb ümber x-telje $\theta_3(y2, y3) - negativ = -30°$
4. Liigend 4 pöörleb ümber y-telje, teljed on kollineaarsed $\theta_4 = 0$

3.1.2 Rotatsioonimaatriksite määramine

Kirjutame rotatsioonimaatriksid. Rotatsioonimaatriksid peavad kajastama eelmise koordinaadistiku pöörlemist nii, et see võtaks järgmise koordinaadistiku orientatsiooni.

Liigendi 1 rotatsioonimaatriks. M_0^1 maatriks

$R_0 \rightarrow R_1$: kuidas pöörata R_0 koordinaadistiku nii, et see oleks orienteeritud nagu R_1 koordinaadistik.

$$M_0^1(x,\theta_1) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\theta_1 & -sin\theta_1 \\ 0 & sin\theta_1 & cos\theta_1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Liigendi 2 rotatsioonimaatriks. M_1^2 maatriks

$R_1 \rightarrow R_2$: kuidas pöörata R_1 koordinaadistiku nii, et see oleks orienteeritud nagu R_2 koordinaadistik.

$$M_1^2(z,\theta_2) = \begin{bmatrix} cos\theta_2 & -sin\theta_2 & 0 \\ sin\theta_2 & cos\theta_2 & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

Liigendi 3 rotatsioonimaatriks. M_2^3 maatriks

$R_2 \rightarrow R_3$: kuidas pöörata R_2 koordinaadistiku nii, et see oleks orienteeritud nagu R_3 koordinaadistik.

$$M_2^3(x,\theta_3) = \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\theta_3 & -sin\theta_3 \\ 0 & sin\theta_3 & cos\theta_3 \end{bmatrix}$$

Liigendi 4 rotatsioonimaatriks. M_3^4 maatriks

$R_3 \rightarrow R_4$: kuidas pöörata R_3 koordinaadistiku nii, et see oleks orienteeritud nagu R_4 koordinaadistik.

$$M_3^4(y,\theta_4) = \begin{bmatrix} cos\theta_4 & 0 & sin\theta_4 \\ 0 & 1 & 0 \\ -sin\theta_4 & 0 & cos\theta_4 \end{bmatrix}$$

Maatriksite orientatsioon R_0 koordinaadistikule kirjeldame nende korrutisena.

$$M_0^4 = M_0^1 M_1^2 M_2^3 M_3^4$$

Autor lahendas selle ülesande maatriksite järjestikuse korrutamise teel.

M_0^2 maatriks

$$M_0^2 = M_0^1 M_1^2 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} cos\theta_2 & -sin\theta_2 & 0 \\ sin\theta_2 & cos\theta_2 & 0 \\ 0 & 0 & 1 \end{bmatrix} = M_1^2$$

$\boldsymbol{M_0^3}$ **maatriks**

$$M_0^3 = M_0^2 M_2^3 = \begin{bmatrix} cos\theta_2 & -sin\theta_2 & 0 \\ sin\theta_2 & cos\theta_2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 \\ 0 & cos\theta_3 & -sin\theta_3 \\ 0 & sin\theta_3 & cos\theta_3 \end{bmatrix}$$

$$= \begin{bmatrix} cos\theta_2 + 0 + 0 & 0 + (-sin\theta_2)cos\theta_3 + 0 & 0 - sin\theta_2(-sin\theta_3) + 0 \\ sin\theta_2 + 0 + 0 & 0 + cos\theta_2 cos\theta_3 + 0 & 0 + cos\theta_2(-sin\theta_3) + 0 \\ 0 + 0 + 0 & 0 + 0 + sin\theta_3 & 0 + 0 + cos\theta_3 \end{bmatrix} =$$

$$= \begin{bmatrix} cos\theta_2 & (-sin\theta_2)cos\theta_3 & -sin\theta_2(-sin\theta_3) \\ sin\theta_2 & cos\theta_2 cos\theta_3 & cos\theta_2(-sin\theta_3) \\ 0 & sin\theta_3 & cos\theta_3 \end{bmatrix}$$

$\boldsymbol{M_0^4}$ **maatriks**

$$M_0^4 = M_0^3 M_3^4 = \begin{bmatrix} cos\theta_2 & (-sin\theta_2)cos\theta_3 & -sin\theta_2(-sin\theta_3) \\ sin\theta_2 & cos\theta_2 cos\theta_3 & cos\theta_2(-sin\theta_3) \\ 0 & sin\theta_3 & cos\theta_3 \end{bmatrix} \cdot \begin{bmatrix} cos\theta_4 & 0 & sin\theta_4 \\ 0 & 1 & 0 \\ -sin\theta_4 & 0 & cos\theta_4 \end{bmatrix} =$$

$$= \begin{bmatrix} cos\theta_2 cos\theta_4 + 0 + (-sin\theta_2)(-sin\theta_3)(-sin\theta_4) & 0 + (-sin\theta_2)cos\theta_3 + 0 & cos\theta_2 sin\theta_4 + 0 + (-sin\theta_2)(-sin\theta_3)cos\theta_4 \\ sin\theta_2 cos\theta_4 + 0 + cos\theta_2(-sin\theta_3)(-sin\theta_4) & 0 + cos\theta_2 cos\theta_3 + 0 & sin\theta_2 sin\theta_4 + 0 + cos\theta_2(-sin\theta_3)cos\theta_4 \\ 0 + 0 + cos\theta_3(-sin\theta_4) & 0 + sin\theta_3 + 0 & 0 + 0 + cos\theta_3 cos\theta_4 \end{bmatrix}$$

$$= \begin{bmatrix} cos\theta_2 cos\theta_4 + (-sin\theta_2)(-sin\theta_3)(-sin\theta_4) & (-sin\theta_2)cos\theta_3 & cos\theta_2 sin\theta_4 + (-sin\theta_2)(-sin\theta_3)cos\theta_4 \\ sin\theta_2 cos\theta_4 + cos\theta_2(-sin\theta_3)(-sin\theta_4) & cos\theta_2 cos\theta_3 & sin\theta_2 sin\theta_4 + cos\theta_2(-sin\theta_3)cos\theta_4 \\ cos\theta_3(-sin\theta_4) & sin\theta_3 & cos\theta_3 cos\theta_4 \end{bmatrix}$$

Eelnevaid ja järgnevaid koordinaadistikute ühendavad vektorid antud manipulaatori puhul saab kirjutada niimoodi:

$$\overrightarrow{OO_1}_{/R_0} = \begin{bmatrix} T_{r1} \\ 0 \\ l_1^z \end{bmatrix}$$

$$\overrightarrow{O_1O_2}_{/R_1} = \begin{bmatrix} 0 \\ l_2^y \\ l_2^z \end{bmatrix}$$

$$\overrightarrow{O_2O_3}_{/R_2} = \begin{bmatrix} 0 \\ l_3 \\ 0 \end{bmatrix}$$

$$\overrightarrow{O_3O_4}_{/R_3} = \begin{bmatrix} 0 \\ l_4 \\ 0 \end{bmatrix}$$

$$\overrightarrow{O_4P}_{/R_4} = \begin{bmatrix} 0 \\ l_5 \\ 0 \end{bmatrix}$$

Kirjutame vektori $\overrightarrow{OP}_{/R_0}$, mis ühendab baaskoordinaadistiku tööriistaga seotud koordinaadistikuga.

Paneme valemisse rotatsiooni maatriksid ja vektorid.

$$\overrightarrow{O_4P}_{/R_0} = \overrightarrow{OO_1}_{/R_0} + M_0^1\overrightarrow{O_1O_2}_{/R_1} + M_0^2\overrightarrow{O_2O_3}_{/R_2} + M_0^3\overrightarrow{O_3O_4}_{/R_3} + M_0^4\overrightarrow{O_4P}_{/R_4} =$$

$$= \begin{bmatrix} cos\theta_2 cos\theta_4 + (-sin\theta_2)(-sin\theta_3)(-sin\theta_4) & (-sin\theta_2)cos\theta_3 & cos\theta_2 sin\theta_4 + (-sin\theta_2)(-sin\theta_3)cos\theta_4 \\ sin\theta_2 cos\theta_4 + cos\theta_2(-sin\theta_3)(-sin\theta_4) & cos\theta_2 cos\theta_3 & sin\theta_2 sin\theta_4 + cos\theta_2(-sin\theta_3)cos\theta_4 \\ cos\theta_3(-sin\theta_4) & sin\theta_3 & cos\theta_3 cos\theta_4 \end{bmatrix} \begin{bmatrix} 0 \\ l_5 \\ 0 \end{bmatrix}$$
$$= \begin{bmatrix} (-sin\theta_2)cos\theta_3 l_5 \\ cos\theta_2 cos\theta_3 l_5 \\ sin\theta_3 l_5 \end{bmatrix}$$

$$\overrightarrow{OP}_{/Ro} = \begin{bmatrix} T_{r1} \\ 0 \\ l_1^z \end{bmatrix} + \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ l_2^y \\ l_2^z \end{bmatrix} + \begin{bmatrix} cos\theta_2 & -sin\theta_2 & 0 \\ sin\theta_2 & cos\theta_2 & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ l_3 \\ 0 \end{bmatrix}$$
$$+ \begin{bmatrix} cos\theta_2 & (-sin\theta_2)cos\theta_3 & -sin\theta_2(-sin\theta_3) \\ sin\theta_2 & cos\theta_2 cos\theta_3 & cos\theta_2(-sin\theta_3) \\ 0 & sin\theta_3 & cos\theta_3 \end{bmatrix} \begin{bmatrix} 0 \\ l_4 \\ 0 \end{bmatrix} +$$

$$+ \begin{bmatrix} cos\theta_2 cos\theta_4 + (-sin\theta_2)(-sin\theta_3)(-sin\theta_4) & (-sin\theta_2)cos\theta_3 & cos\theta_2 sin\theta_4 + (-sin\theta_2)(-sin\theta_3)cos\theta_4 \\ sin\theta_2 cos\theta_4 + cos\theta_2(-sin\theta_3)(-sin\theta_4) & cos\theta_2 cos\theta_3 & sin\theta_2 sin\theta_4 + cos\theta_2(-sin\theta_3)cos\theta_4 \\ cos\theta_3(-sin\theta_4) & sin\theta_3 & cos\theta_3 cos\theta_4 \end{bmatrix} \begin{bmatrix} 0 \\ l_5 \\ 0 \end{bmatrix}$$

$$= \begin{bmatrix} T_{r1} \\ 0 \\ l_1^z \end{bmatrix} + \begin{bmatrix} 0 \\ l_2^y \\ l_2^z \end{bmatrix} + \begin{bmatrix} -sin\theta_2 l_3 \\ cos\theta_2 l_3 \\ 0 \end{bmatrix} + \begin{bmatrix} (-sin\theta_2)cos\theta_3 l_4 \\ cos\theta_2 cos\theta_3 l_4 \\ sin\theta_3 l_4 \end{bmatrix} + \begin{bmatrix} (-sin\theta_2)cos\theta_3 l_5 \\ cos\theta_2 cos\theta_3 l_5 \\ sin\theta_3 l_5 \end{bmatrix} =$$

$$= \begin{bmatrix} T_{r1} + (-sin\theta_2 l_3) + (-sin\theta_2)cos\theta_3 l_4 + (-sin\theta_2)cos\theta_3 l_5 \\ l_2^y + cos\theta_2 l_3 + cos\theta_2 cos\theta_3 l_4 + cos\theta_2 cos\theta_3 l_5 \\ l_1^z + l_2^z + sin\theta_3 l_4 + sin\theta_3 l_5 \end{bmatrix} =$$

$$= \begin{bmatrix} T_{r1} - sin\theta_2 l_3 - sin\theta_2 cos\theta_3 l_4 - sin\theta_2 cos\theta_3 l_5 \\ l_2^y + cos\theta_2 l_3 + cos\theta_2 cos\theta_3 l_4 + cos\theta_2 cos\theta_3 l_5 \\ l_1^z + l_2^z + sin\theta_3 l_4 + sin\theta_3 l_5 \end{bmatrix} =$$

$$= \begin{bmatrix} T_{r1} - sin\theta_2(l_3 + cos\theta_3 l_4 + cos\theta_3 l_5) \\ l_2^y + cos\theta_2(l_3 + cos\theta_3 l_4 + cos\theta_3 l_5) \\ l_1^z + l_2^z + sin\theta_3(l_4 + l_5) \end{bmatrix} = \begin{bmatrix} T_{r1} - sin\theta_2(l_3 + cos\theta_3(l_4 + l_5)) \\ l_2^y + cos\theta_2(l_3 + cos\theta_3(l_4 + l_5)) \\ l_1^z + l_2^z + sin\theta_3(l_4 + l_5) \end{bmatrix}$$

$$\overrightarrow{OP}_{/Ro} = \begin{bmatrix} T_{r1} - sin\theta_2(l_3 + cos\theta_3(l_4 + l_5)) \\ l_2^y + cos\theta_2(l_3 + cos\theta_3(l_4 + l_5)) \\ l_1^z + l_2^z + sin\theta_3(l_4 + l_5) \end{bmatrix}$$

3.1.3 Lõppefektori koordinatide määramine

Saadud vektor kujutab tööriista koordinaadid põhisüsteemis.

$$\overrightarrow{OP}_{/Ro} = \begin{bmatrix} T_{r1} - sin\theta_2(l_3 + cos\theta_3(l_4 + l_5)) \\ l_2^y + cos\theta_2(l_3 + cos\theta_3(l_4 + l_5)) \\ l_1^z + l_2^z + sin\theta_3(l_4 + l_5) \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \end{bmatrix}$$

Siit saame väljendada valemi koordinaatide arvutamiseks:

$x = T_{r1} - sin\theta_2(l_3 + cos\theta_3 \cdot (l_4 + l_5))$

$y = l_2^y + cos\theta_2(l_3 + cos\theta_3 \cdot (l_4 + l_5))$

$z = l_1^z + l_2^z + sin\theta_3 \cdot (l_4 + l_5)$

Lihtsustame valemi:

$x = T_{r1} - sin\theta_2(0{,}25 + cos\theta_3)$

$y = 0{,}25 + cos\theta_2(0{,}25 + cos\theta_3)$

$z = 1 + sin\theta_3$

Lähteandmetel on tähistatud manipulaatori geometriliste parameetrid: nurgad ja distantsid.

$\theta_3 = -30°$

l_2^y=0,25 m

$l_3 = 0{,}25\ \mathrm{m}$

$l_2^z = 0{,}75\ \mathrm{m}$

$l_4 + l_5 = 1\ \mathrm{m}$

Autor võttis teised geomeetrilised parameetrid, mis põhinevad manipulaatori konstruktsioonil ja proportsioonidel.

$\theta_2 = -45°$

$T_{r1} = 0{,}25\ \mathrm{m}$

$l_1^z = 0{,}25\ \mathrm{m}$

Edasi autor arvutas haaratsi koordinaadid baaskoordinaadistikus, asendades valemis konkreetsed väärtused.

$$x = T_{r1} - sin\theta_2(l_3 + cos\theta_3 \cdot (l_4 + l_5)) = 0{,}25 - \sin(-45°)(0{,}25 + \cos(-30°) \cdot 1) =$$
$$= 0{,}25 - \left(-\frac{\sqrt{2}}{2}\right)\left(0{,}25 + \frac{\sqrt{3}}{2} \cdot 1\right) = 0{,}25 + \frac{\sqrt{2}}{2}\left(0{,}25 + \frac{\sqrt{3}}{2}\right) \approx 1{,}039\ \mathrm{m}$$

$$y = l_2^y + cos\theta_2(l_3 + cos\theta_3(l_4 + l_5)) = 0{,}25 + \cos(-45°)\,(0{,}25 + \cos(-30°) \cdot 1) =$$
$$= 0{,}25 + \frac{\sqrt{2}}{2}\left(0{,}25 + \frac{\sqrt{3}}{2}\right) \approx 1{,}039\ \mathrm{m}$$

$$z = l_1^z + l_2^z + sin\theta_3(l_4 + l_5) = 0{,}25 + 0{,}75 + \sin(-30°) \cdot 1 = 1 - 0{,}5 \cdot 1 = 0{,}5 \text{ m}$$

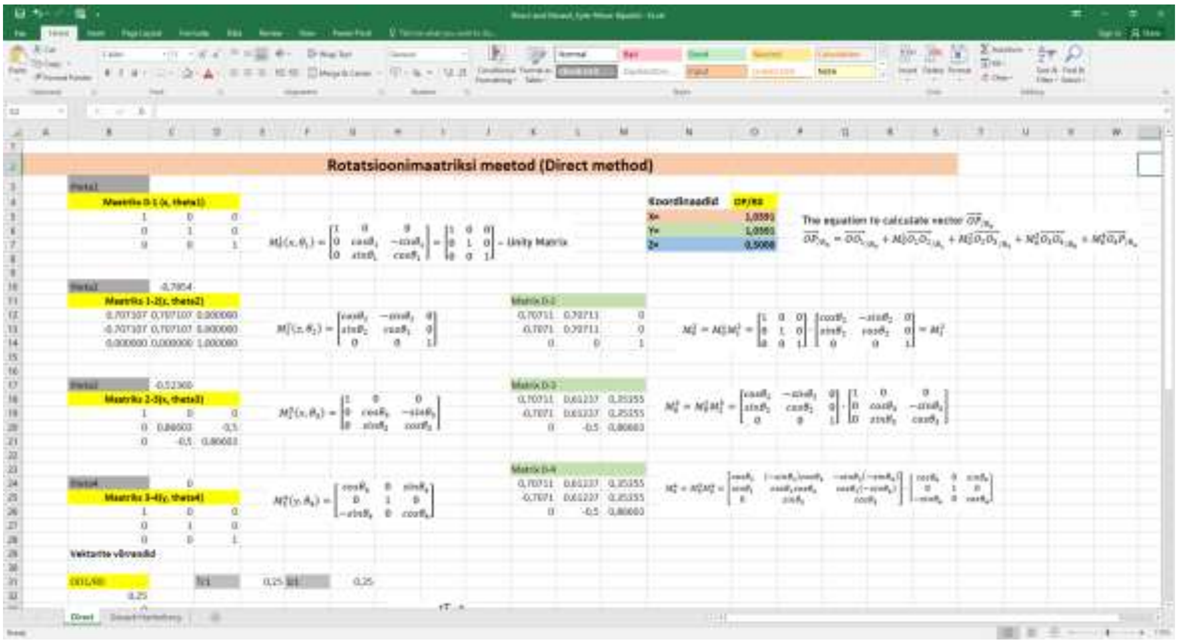
Lõpplahendusena sai autor lõppefektori konkreetsed koordinaadid baaskoordinaadisüsteemis.

$$\overrightarrow{OP}_{/Ro} = \begin{bmatrix} 1{,}039 \\ 1{,}039 \\ 0{,}5 \end{bmatrix}$$

3.1.4 Metoodiline materjal "Rotatsioonimaatriksid"

Arvutuste põhjal koostas autor Excel'is faili "Rotatsioonimaatriksi meetod" (vt Lisa 6 ja Joonis 3.3). Faili abil saab arvutada teiste algväärtustega üldistatud koordinaadid. Sarnasel eesmärgil autor kirjutas koodi programmis Matlab (vt Lisa 2). Sobivate muutujate sisestamisel saab arvutada koodi abil üldistatud koordinaatide väärtused.

Neid faile (vt Lisad 2 ja 6) soovitab autor kasutada üliõpilastele õppevahendina. Nende abil saab, autori meeles, rotatsioonimaatriksi meetodil kiiresti arvutada otsese kinemaatika ülesande, iga etapi visuaalselt demonstreerida ja ka arvutustulemusi kontrollida.



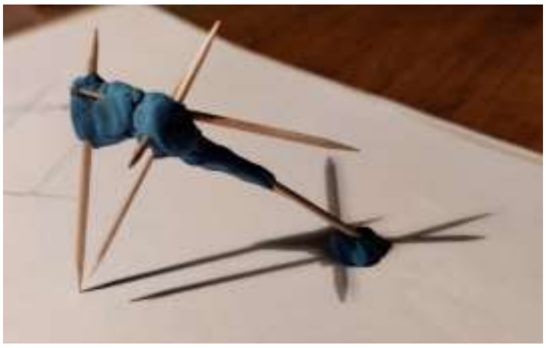
Joonis 3.3 Excel-fail. Rotatsioonimaatriksi meetod

3.2 Denavit-Hartenbergi meetod

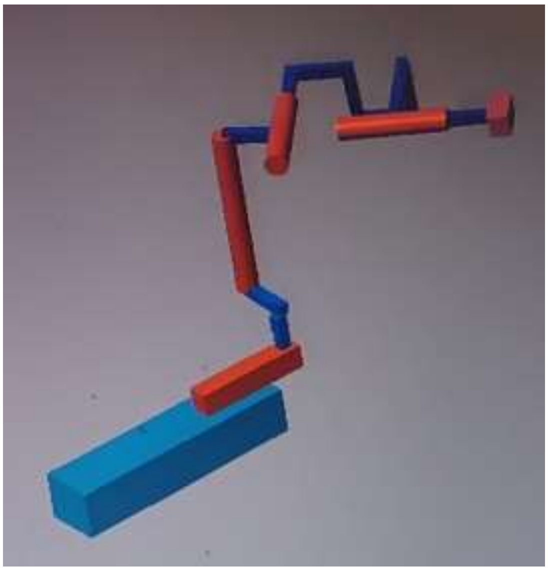
3.2.1 Teljestiku määramine

Esiteks on vaja määrata teljestiku, võttes arvesse lõikes 2.2.1 kirjeldatud reegleid.

Telgede õige suuna määramine võib olla keeruline. Manipulaatori disaini õigeks mõistmiseks autor ehitas esiteks olemasolevatest vahenditest füüsiline prototüüp (vt Joonis 3.4). Seejärel ehitas autor Simscape'is virtuaalse mudeli (vt Joonis 3.5) , mida ta kasutas koordinaatide telgede visualiseerimiseks.



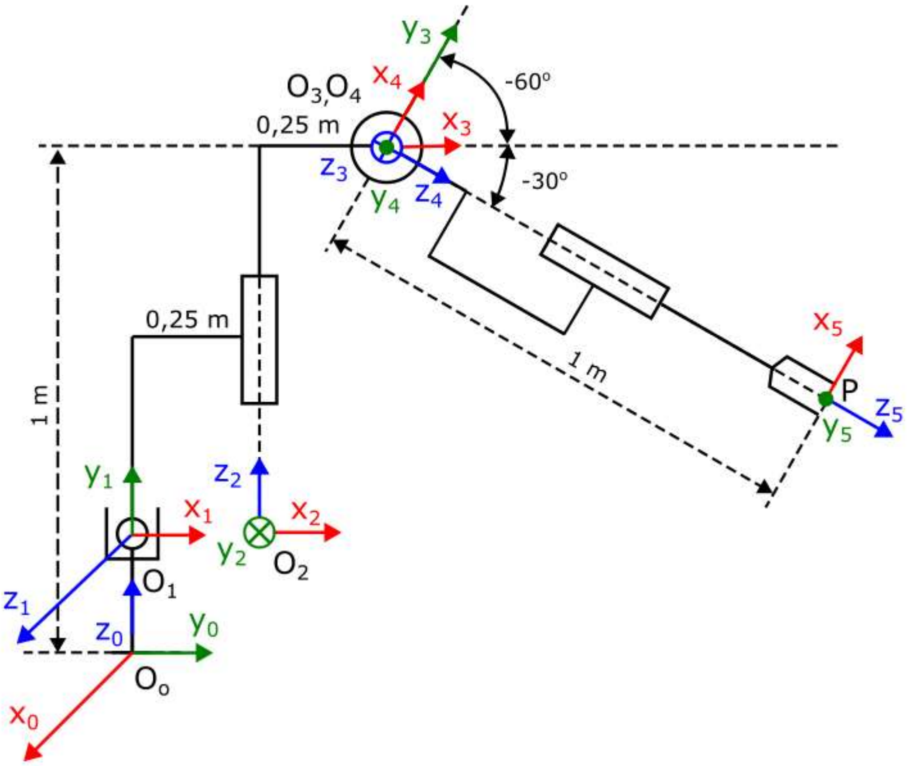
Joonis 3.4 Esimene prototüüp



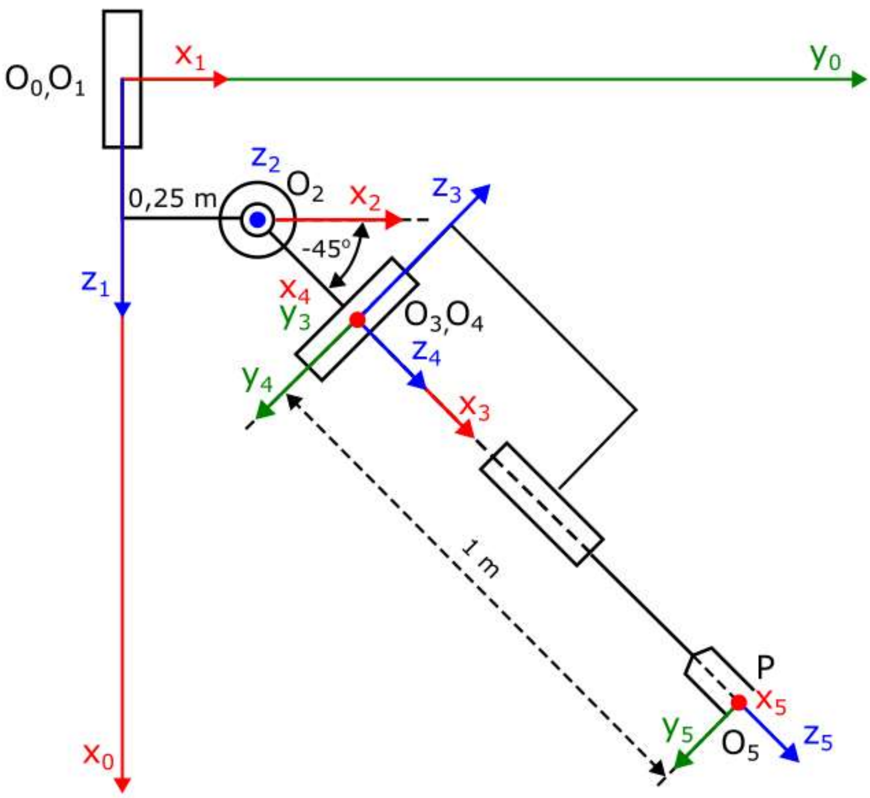
Joonis 3.5 Virtuaalne mudel

Autor korraldas koordinaatsüsteemid nii, et saada võimalikult palju nullväärtusi, järgides samas meetodi piiranguid.

Telgede asukoht on näidatud joonisel 3.6, 3.7.



Joonis 3.6 Denavit-Hartenberg method. Telgede asukoht. Külgvaade



Joonis 3.7 Denavit-Hartenberg method. Teljed. Pealtvaade

3.2.2 Denavit-Hartenbergi parameetrite määramine

Denavit-Hartenbergi parameetrid sisestas autor tabelisse (tabel 3.1).

Tabel 3.1 TRRR-manipulaatori Denavit-Hartenbergi parameetrid

i ($T_{i-1} \rightarrow T_i$)	θ_i	d_i	a_i	α_i
i_1 ($T_0 \rightarrow T_1$)	θ_1	d_1	0	α_1
i_2 ($T_1 \rightarrow T_2$)	θ_2	d_2	a_2	α_2
i_3 ($T_2 \rightarrow T_3$)	θ_3	d_3	a_3	α_3
i_4 ($T_3 \rightarrow T_4$)	θ_4	0	0	α_4
I_5 ($T_4 \rightarrow T_5$)	θ_5	1	0	α_5

Geomeetrilised mõõtmed - nihked autor võttis manipulaatori konstruktsioonist (vt lähteandmeid). Koordinaatide telgede pöördenurgad autor määras, võttes positiivse suunana arvesse vastupäeva, negatiivset – päripäeva. Autor sisestas tabelisse 3.2 parameetrite tegelikud väärtused.

Tabel 3.2. Denavit-Hartenbergi parameetrid. Tegelikud väärtused

i ($T_{i-1} \rightarrow T_i$)	θ_i	d_i	a_i	α_i
i_1 ($T_0 \rightarrow T_1$)	$90°$	0,25	0	$90°$
i_2 ($T_1 \rightarrow T_2$)	$0°$	0,25	0,25	$-90°$
i_3 ($T_2 \rightarrow T_3$)	$-45°$	0,75	0,25	$-90°$
i_4 ($T_3 \rightarrow T_4$)	$-60°$	0	0	$-90°$
I_5 ($T_4 \rightarrow T_5$)	$0°$	1	0	0

3.2.3 Teisendusmaatriksite määramine. Üldvalem

Teisendusmaatriksid T_0^1, T_2^1, T_3^2, T_4^3 leiakse valemiga:

$$T_i^{i-1} = \begin{bmatrix} cos\theta_i & -sin\theta_i cos\alpha_i & sin\theta_i sin\alpha_i & a_i cos\theta_i \\ sin\theta_i & cos\theta_i cos\alpha_i & -cos\theta_i sin\alpha_i & a_i sin\theta_i \\ 0 & S\alpha_i & cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Asendades teadaolevad nurgaväärtused, autor sai viis homogeensete teisenduste maatriksit, kus kasutas järgmist tähistust: $C_i = cos\theta_i$, $S_i = sin\theta_i$.

$$T_1^0 = \begin{bmatrix} C_1 & 0 & S_1 & 0 \\ S_1 & 0 & -C_1 & 0 \\ 0 & 1 & 0 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad T_2^1 = \begin{bmatrix} C_2 & 0 & -S_2 & a_2C_2 \\ S_2 & 0 & C_2 & a_2S_2 \\ 0 & -1 & 0 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad T_3^2 = \begin{bmatrix} C_3 & 0 & -S_3 & a_3C_3 \\ S_3 & 0 & C_3 & a_3S_3 \\ 0 & -1 & 0 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$T_4^3 = \begin{bmatrix} C_4 & 0 & -S_4 & 0 \\ S_4 & 0 & C_4 & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \qquad T_4^5 = \begin{bmatrix} C_5 & -S_5 & 0 & 0 \\ S_5 & C_5 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Maatriksite järjestikuse korrutamisel sai autor manipulaatori lõpliku maatriksi.

$$T_5^0 = T_1^0T_2^1T_3^2T_4^3T_5^4 = \begin{bmatrix} n_x & s_x & a_x & p_x \\ n_y & s_y & a_y & p_y \\ n_z & s_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} n_0^5 & s_0^5 & a_0^5 & p_0^5 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} R_0^5 & p_0^5 \\ 0 & 1 \end{bmatrix}$$

Kus

$$n_x = C_1C_2(C_3C_4C_5 + S_3S_5) - S_1(S_3C_4C_5 - C_3S_5) + C_1S_2S_4C_5$$

$$n_y = S_1C_2(C_3C_4C_5 + S_3S_5) + C_1(S_3C_4C_5 - C_3S_5) + S_1S_2S_4C_5$$

$$n_z = S_2(C_3C_4C_5 + S_3S_5) - C_2S_4C_5$$

$$s_x = C_1C_2(-C_3C_4S_5 + S_3C_5) + S_1(S_3C_4S_5 - C_3C_5) - C_1S_2S_4S_5$$

$$s_y = S_1C_2(-C_3C_4S_5 + S_3C_5) - C_1(S_3C_4S_5 + C_3C_5) - S_1S_2S_4S_5$$

$$s_z = S_2(-C_3C_4S_5 + S_3C_5) + C_2S_4S_5$$

$$a_x = -C_1C_2C_3S_4 + S_1S_3S_4 + C_1S_2C_4$$

$$a_y = -S_1C_2C_3S_4 - C_1S_3S_4 + S_1S_2C_4$$

$$a_z = -S_2C_3S_4 - C_2C_4$$

$$p_x = C_1C_2(-C_3S_4 + a_3C_3) + S_1(S_3S_4 - a_3S_3) - C_1S_2(-C_4 + d_3) + a_2C_1C_2 + d_2S_1$$

$$p_y = S_1C_2(-C_3S_4 + a_3C_3) + C_1(-S_3S_4 - a_3S_3) - S_1S_2(-C_4 + d_3) + a_2S_1C_2 - d_2S_1$$

$$p_z = S_2(-C_3S_4 + a_3C_3) + C_2(-C_4 + d_3) + a_2S_2 + d_1$$

Vektorid n_0^5 s_0^5 a_0^5 mõõtmetega 1x3 need on vektorid x_5, y_5, z_5 baaskoordinaadistiku $o_ox_0y_0z_0$ suhtes.

p_0^5 on süsteemi alguspunkti lineaarne nihkevektor $o_5x_5y_5z_5$ koordinaadistiku $o_ox_0y_0z_0$ suhtes.

Matriks R_0^5 mõõtmetega 3x3 - see on pöördmaatriks $o_5x_5y_5z_5$ koordinaadistiku $o_ox_0y_0z_0$ suhtes.

Matemaatilisi arvutusi saab üksikasjalikumalt näha lisas 1.

3.2.4 Teisendusmaatriksite arvutamine

Asendades lähteandmetest võetud arvväärtusi, tegi autor vastavad arvutused. Arvutuste tulemuseks oli kindlate arvväärtustega teisendusmaatriks. Maatriksi parem pool kujutab haaratsi koordinaate baaskoordinaadisüsteemis.

$$\begin{aligned} n_x = C_1C_2(C_3C_4C_5 + S_3S_5) - S_1(S_3C_4C_5 - C_3S_5) + C_1S_2S_4C_5 \\ = cos90°cos0°(\cos(-45°)\cos(-60°)\,cos0° + \sin(-45°)\,sin0°) \\ - sin90°(\sin(-45°)\cos(-60°)\,cos0° - \cos(-45°)sin0°) \\ + cos90°sin0°sin(-60°)cos0° = -(1)\left(\frac{-\sqrt{2}}{2}\right)\cdot\frac{1}{2}\cdot 1 = \frac{\sqrt{2}}{4} \end{aligned}$$

$$\begin{aligned} n_y = S_1C_2(C_3C_4C_5 + S_3S_5) + C_1(S_3C_4C_5 - C_3S_5) + S_1S_2S_4C_5 \\ = sin90°cos0°(\cos(-45°)\cos(-60°)\,cos0° + \sin(-45°)\,sin0°) \\ + cos90°(\sin(-45°)\cos(-60°)\,cos0° - \cos(-45°)sin0°) \\ + sin90°sin0°sin(-60°)cos0° = 1\cdot 1\cdot\frac{\sqrt{2}}{2}\cdot\frac{1}{2}\cdot 1 = \frac{\sqrt{2}}{4} \end{aligned}$$

$$\begin{aligned} n_z = S_2(C_3C_4C_5 + S_3S_5) - C_2S_4C_5 \\ = sin0°(\cos(-45°)\cos(-60°)\,cos0° + \sin(-45°)\,sin0°) \\ - cos0°sin(-60°)cos0° = \frac{\sqrt{3}}{2} \end{aligned}$$

$$\begin{aligned} s_x = C_1C_2(-C_3C_4S_5 + S_3C_5) + S_1(S_3C_4S_5 - C_3C_5) - C_1S_2S_4S_5 \\ = cos90°cos0°(-\cos(-45°)\cos(-60°)\,sin0° + \sin(-45°)\,cos0°) \\ + sin90°(\sin(-45°)\cos(-60°)\,sin0° - \cos(-45°)cos0°) \\ - cos90°sin0°sin(-60°)sin0° = \frac{\sqrt{2}}{2} \end{aligned}$$

$$\begin{aligned} s_y = S_1C_2(-C_3C_4S_5 + S_3C_5) - C_1(S_3C_4S_5 + C_3C_5) - S_1S_2S_4S_5 = \\ = sin90°cos0°(-\cos(-45°)\cos(-60°)\,sin0° + \sin(-45°)\,cos0°) \\ - cos90°(\sin(-45°)\cos(-60°)\,sin0° + \cos(-45°)cos0°) \\ - sin90°sin0°sin(-60°)sin0° = \frac{-\sqrt{2}}{2} \end{aligned}$$

$$\begin{aligned} s_z = S_2(-C_3C_4S_5 + S_3C_5) + C_2S_4S_5 \\ = sin0°(-\cos(-45°)\cos(-60°)\,sin0° + \sin(-45°)\,cos0°) \\ + cos0°sin(-60°)sin0° = 0 \end{aligned}$$

$$\begin{aligned} a_x = -C_1C_2C_3S_4 + S_1S_3S_4 + C_1S_2C_4 \\ = -cos90°cos0°\cos(-45°)sin(-60°) + sin90°\sin(-45°)\,sin(-60°) \\ + cos90sin0°\cos(-60°) = \frac{\sqrt{2}\sqrt{3}}{4} \end{aligned}$$

$$\begin{aligned} a_y = -S_1C_2C_3S_4 - C_1S_3S_4 + S_1S_2C_4 \\ = -sin90°cos0°\cos(-45°)sin(-60°) - cos90°\sin(-45°)\,sin(-60°) \\ + sin90°sin0°\cos(-60°) = \frac{\sqrt{2}\sqrt{3}}{4} \end{aligned}$$

$$a_z = -S_2C_3S_4 - C_2C_4 = -sin0°\cos(-45°)sin(-60°) - cos0°\cos(-60°) = -\frac{1}{2}$$

$$\begin{aligned}p_x = C_1C_2(-C_3S_4 + a_3C_3) + S_1(S_3S_4 - a_3S_3) - C_1S_2(-C_4 + d_3) + a_2C_1C_2 + d_2S_1 \\ = cos90°cos0°\big(-\cos(-45°)sin(-60°) + 0{,}25\cos(-45°)\big) \\ + sin90°\big(\sin(-45°)\,sin(-60°) - 0{,}25sin(-45°)\big) \\ - cos90°sin0°(-sin(-60°) + 0{,}75) + 0{,}25cos90°cos0° + 0{,}25sin90° \\ = \frac{\sqrt{2}\sqrt{3}}{4} + 0{,}25\cdot\left(\frac{\sqrt{2}}{2} + 1\right)\end{aligned}$$

$$\begin{aligned}p_y = S_1C_2(-C_3S_4 + a_3C_3) + C_1(-S_3S_4 - a_3S_3) - S_1S_2(-C_4 + d_3) + a_2S_1C_2 - d_2S_1 = \\ = sin90°cos0°\big(-\cos(-45°)sin(-60°) + 0{,}25\cos(-45°)\big) \\ + cos90°\big(-\sin(-45°)\,sin(-60°) - 0{,}25sin(-45°)\big) \\ - sin90°sin0°(-cos(-60°) + 0{,}75) + 0{,}25sin90°cos0° - 0{,}25sin90° \\ = \frac{\sqrt{2}\sqrt{3}}{4} + 0{,}25\cdot\left(\frac{\sqrt{2}}{2} + 1\right)\end{aligned}$$

$$\begin{aligned}p_z = S_2(-C_3S_4 + a_3C_3) + C_2(-C_4 + d_3) + a_2S_2 + d_1 \\ = sin0°(-\cos(-45°)sin(-60°) + 0{,}25\cos(-45°)) + cos0°(-cos(-60°) \\ + 0{,}75) + 0{,}25sin0° + 0{,}25 = \frac{1}{2}\end{aligned}$$

Lõpuks saime:

$$T_5^0 = T_1^0T_2^1T_3^2T_4^3T_5^4 = \begin{bmatrix} n_x & s_x & a_x & p_x \\ n_y & s_y & a_y & p_y \\ n_z & s_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} \frac{\sqrt{2}}{4} & \frac{\sqrt{2}}{2} & \frac{\sqrt{2}\sqrt{3}}{4} & \frac{\sqrt{2}\sqrt{3}}{4} + 0{,}25\cdot\left(\frac{\sqrt{2}}{2}+1\right) \\ \frac{\sqrt{2}}{4} & -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}\sqrt{3}}{4} & \frac{\sqrt{2}\sqrt{3}}{4} + 0{,}25\cdot\left(\frac{\sqrt{2}}{2}+1\right) \\ \frac{\sqrt{3}}{2} & 0 & -\frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 0 & 1 \end{bmatrix} \approx$$

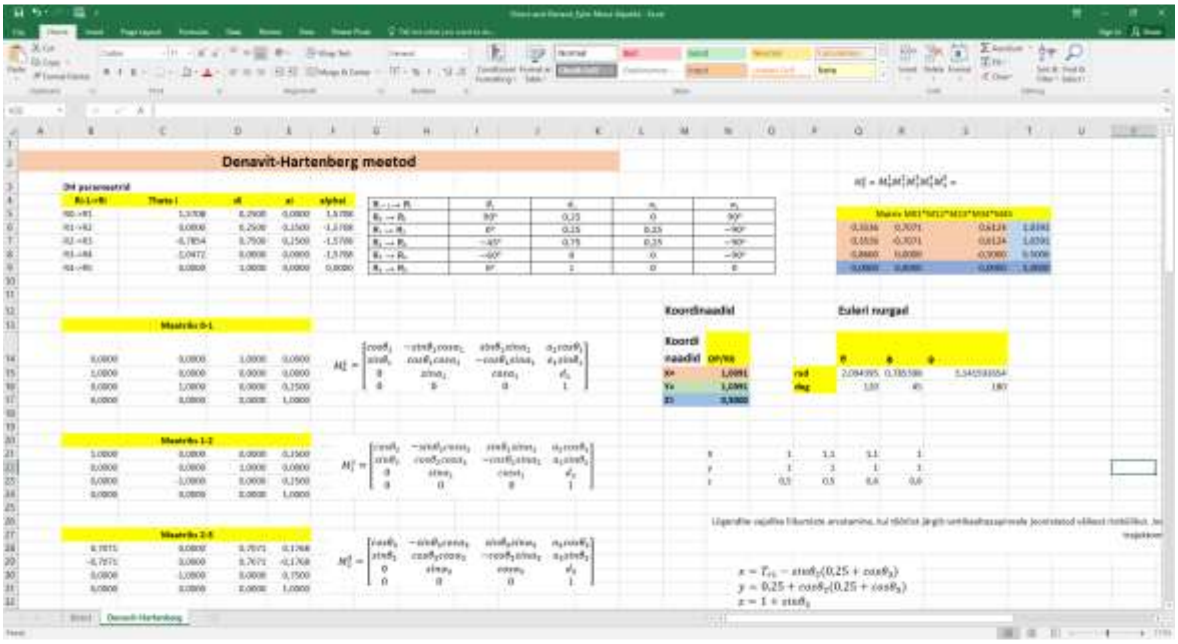
$$\approx \begin{bmatrix} 0{,}3536 & 0{,}7071 & 0{,}6124 & 1{,}0391 \\ 0{,}3536 & -0{,}7071 & 0{,}6124 & 1{,}0391 \\ 0{,}8660 & 0 & -0{,}5 & 0{,}5 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Siit saame manipulaatori tööorgani koordinaadid.

$$\overrightarrow{OP}_{/Ro} = \begin{bmatrix} 1{,}039 \\ 1{,}039 \\ 0{,}5 \end{bmatrix}$$

3.2.5 Metoodiline materjal "Denavit-Hartenbergi meetod"

Autor kasutas ülalkirjeldatud valemeid üldkujul Excel'i failis "Denavit-Hartenberg" (vt Joonis 3.8, Lisa 6). Denavit-Hartenbergi parameetrite erinevate väärtuste asendamisega saab saada teisendusmaatriksi kuju teatud manipulaatori konstruktsiooni puhul.



Joonis 3.8. Excel-fail. Denavit-Hartenbergi meetod

Samuti leidis autor, et teisendusmaatriksi käsitsi arvutamiseks on mugavam numbrilised väärtused koheselt asendada. Maatriksite ja nende korrutamise arvutamine saab näha Lisas 3.

Arvutuste kiirendamiseks ja simulatsioonide loomiseks võib kasutada Matlab'i programmi. Autor kasutas programmi võimalusi, et luua kood, mis arvutab teisendusmaatriksi ja ehitab erinevaid manipulaatori simulatsioone. Programmi lähteandmed on Denavit-Hartenbergi parameetrid (vt Lisa 4 "Denavit-Hartenberg. Kood"). Lisaks kasutas autor manipulaatori ja selle liikumise visualiseerimiseks programmi Robotics Toolbox (vt Lisa 5).

3.3 Euleri nurkade arvutamine

Üldistatud rotatsioonimaatriksi valem

$$R_{z\times z} = R_{z,\psi} \cdot R_{x,\theta} \cdot R_{z,\varphi} = \begin{bmatrix} cos\psi cos\varphi - cos\theta sin\psi sin\varphi & -cos\varphi sin\psi - cos\psi cos\theta sin\psi & sin\theta sin\varphi \\ cos\theta cos\varphi sin\psi + cos\psi sin\varphi & cos\psi cos\theta cos\psi - sin\psi sin\varphi & -cos\varphi sin\theta \\ sin\psi sin\theta & cos\psi sin\theta & cos\theta \end{bmatrix}$$

Autor kirjatas rotatsioonimaatriksi ja asendas selle punktis 3.2 leitud väärtused.

$$R_{z\times z} = R_{z,\psi} \cdot R_{x,\theta} \cdot R_{z,\varphi} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} = \begin{bmatrix} \frac{\sqrt{2}}{4} & \frac{\sqrt{2}}{2} & \frac{\sqrt{2}\sqrt{3}}{4} \\ \frac{\sqrt{2}}{4} & -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}\sqrt{3}}{4} \\ \frac{\sqrt{3}}{2} & 0 & -\frac{1}{2} \end{bmatrix} = \begin{bmatrix} 0{,}3536 & 0{,}7071 & 0{,}6124 \\ 0{,}3536 & -0{,}7071 & 0{,}6124 \\ 0{,}8660 & 0 & -0{,}5 \end{bmatrix}$$

Euleri nurgad on leitavad koordinaatsüsteemist, milles nurk on vahemikus [-90°, 90°].

1. Leiame pöördenurga θ võttes tingimusteks et $r_{33} \neq \pm 1, r_{33} \neq 0$

$$r_{33} = cos\theta = -\frac{1}{2}$$

$$sin\theta = \pm\frac{\sqrt{3}}{2}$$

$$\theta = \arccos\left(-\frac{1}{2}\right) = 120°$$

$\theta = -120°$ - ei sobi Euleri koordinaatsüsteemile

2. Leiame kallutusnurga φ

$$r_{13} = cos\varphi sin\theta = \frac{\sqrt{2}\sqrt{3}}{4} \approx 0{,}6124 \neq 0$$

$$r_{23} = sin\varphi sin\theta = \frac{\sqrt{2}\sqrt{3}}{4} \approx 0{,}6124 \neq 0$$

$$tg\varphi = \frac{r_{13}}{r_{23}} = \frac{\frac{\sqrt{2}\sqrt{3}}{4}}{\frac{\sqrt{2}\sqrt{3}}{4}} = 1$$

$$\varphi = arctg(1) = 45°$$

3. Leiame lengerdusnurga ψ

$$r_{31} = -cos\psi sin\theta = \frac{\sqrt{3}}{2} \approx 0{,}866 \neq 0$$

$$r_{32} = sin\psi sin\theta = 0$$

$$sin\theta = \frac{\sqrt{3}}{2} \neq 0$$

$$sin\psi sin\theta = 0 => sin\psi = 0$$

$$-cos\psi sin\theta = \frac{\sqrt{3}}{2}$$

$$sin\theta = \frac{\sqrt{3}}{2} => -cos\psi\frac{\sqrt{3}}{2} = \frac{\sqrt{3}}{2} => -cos\psi = 1 => cos\psi = -1 => \psi = 180°$$

Antud tingimustel autor sai sellised Euleri nurgad:

$\theta = 120°$ $\varphi = 45°$ $\psi = 180°$

Euleri nurkade arvutamise valem autor sisestas Excel'i faili (vt Lisa 6). Valemite abiga saab arvutada Euleri nurgad muude lähteparameetrite jaoks.

4 MATLAB'I KOOD JA SIMULATSIOON

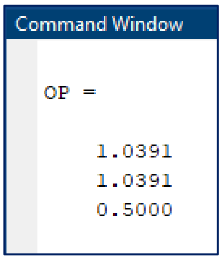
4.1 Rotatsioonimaatriksi meetod. Matlab

Arvutuste lihtsustamiseks ja algoritmi universaalsemaks muutmiseks kasutas autor mitmeid digitaalseid keskkondi.

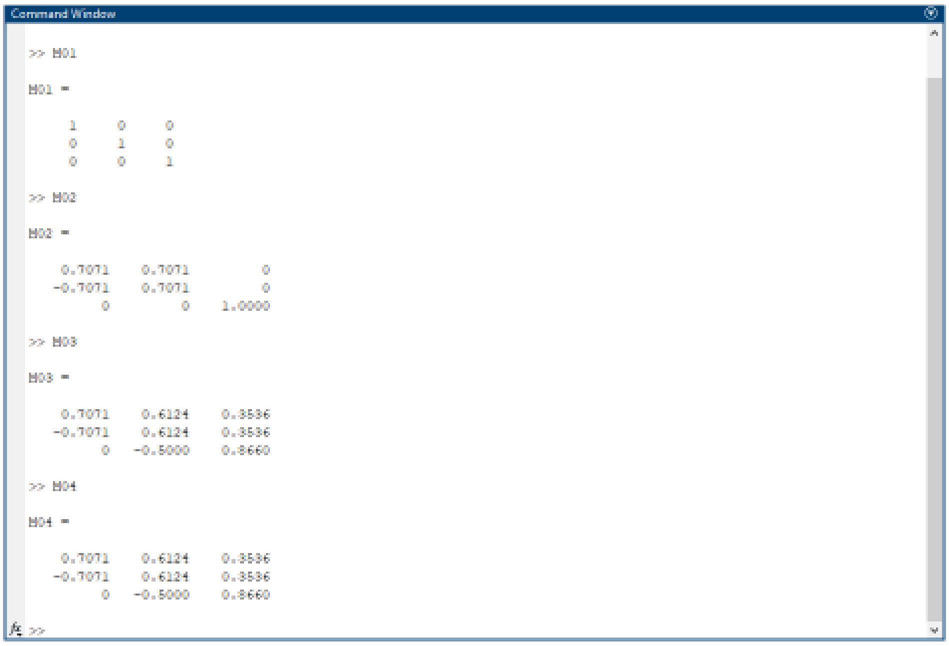
Matemaatilise mudeli saab arvutada Excel'i ja Matlab'i abil. Excel'i keskkonnas esitas autor otsese kinemaatika ülesande lahenduse rotatsioonimaatriksi meetodil, samuti Denavit-Hartenbergi meetodil (vt Lisa 6, Lisa 7).

Autor esitas rotatsioonimaatriksi meetodi Matlab'i koodi kujul [17]. Kood on esitatud Lisas 3.

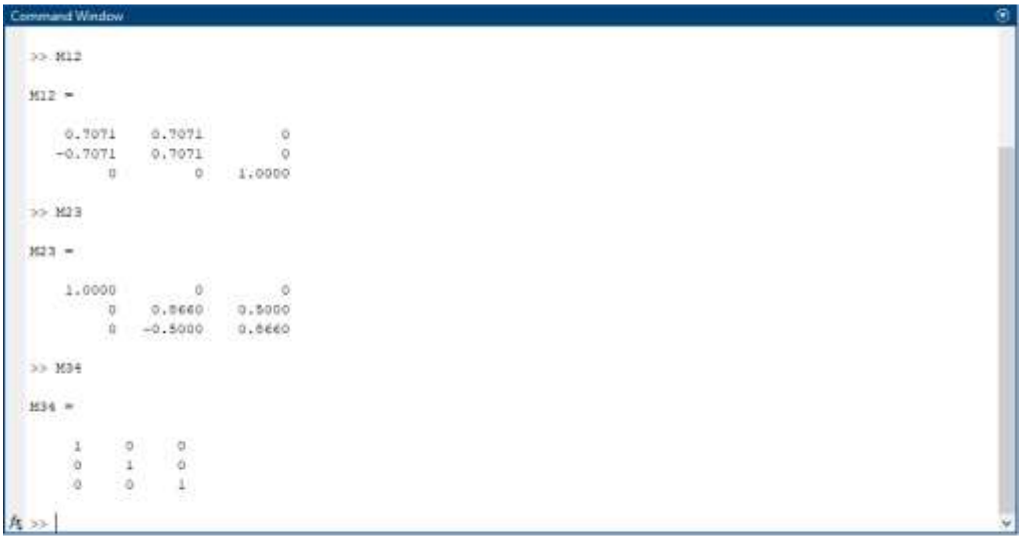
Tänu Matlab'i koodile on võimalik saada lõppefektori koordinaadid põhikoordinaatsüsteemis erinevate algparameetrite määramisel (Joonis 4.1). Samuti saab teha vahearvutusi: näiteks arvutada iga liigendi jaoks pöördemaatriksid ja teisendusmaatriksid (Joonis 4.2, 4.3).



Joonis 4.1 Rotatsioonimaatriksi meetod. Koordinadid baaskoordinadistikus

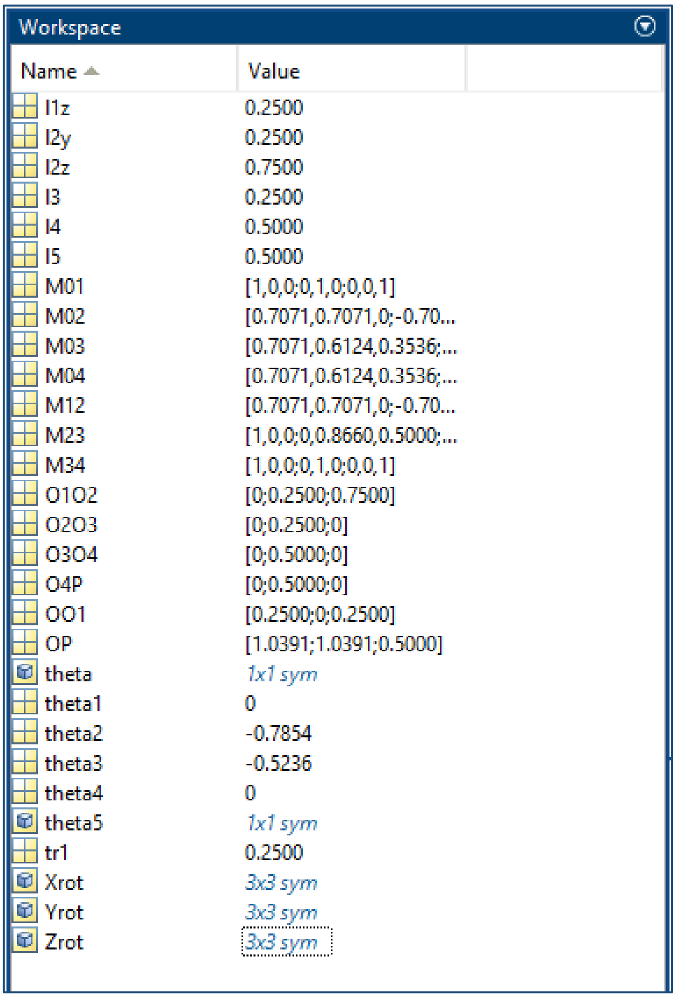


Joonis 4.2 Rotatsiooni maatriksi meetod. Vahetulemused



Joonis 4.3 Rotatsioonimaatriksi meetod. Teisendusmaatriksid. Vahetulemused

Parameetrid, mille väärtusi saab selle koodi abil kuvada, on näidatud joonisel 4.4.



Joonis 4.4 Arvutatud väärtused ja algmuutujad

See kood autori sõnul saab kohandanda algväärtusi muutes ka teiste struktuuritega manipulaatorite arvutamiseks.

Autor kirjeldas allpool koodi põhimõtet.

Pöördmaatriksite valem on määratud iga koordinaattelje jaoks, samuti on määratud manipulaatori konstruktsiooni nihkevektori väärtused. See hõlmab lapse koordinaatsüsteemi pöördenurga väärtuste määramist vanema suhtes. Autor kirjutas programmi asukohavektori valemi. Selle ja eelnevalt määratletud pöördmaatriksite ja nihkevektorite abil arvutatakse manipulaatori haaratsi koordinaadid baaskoordinaatide süsteemis.

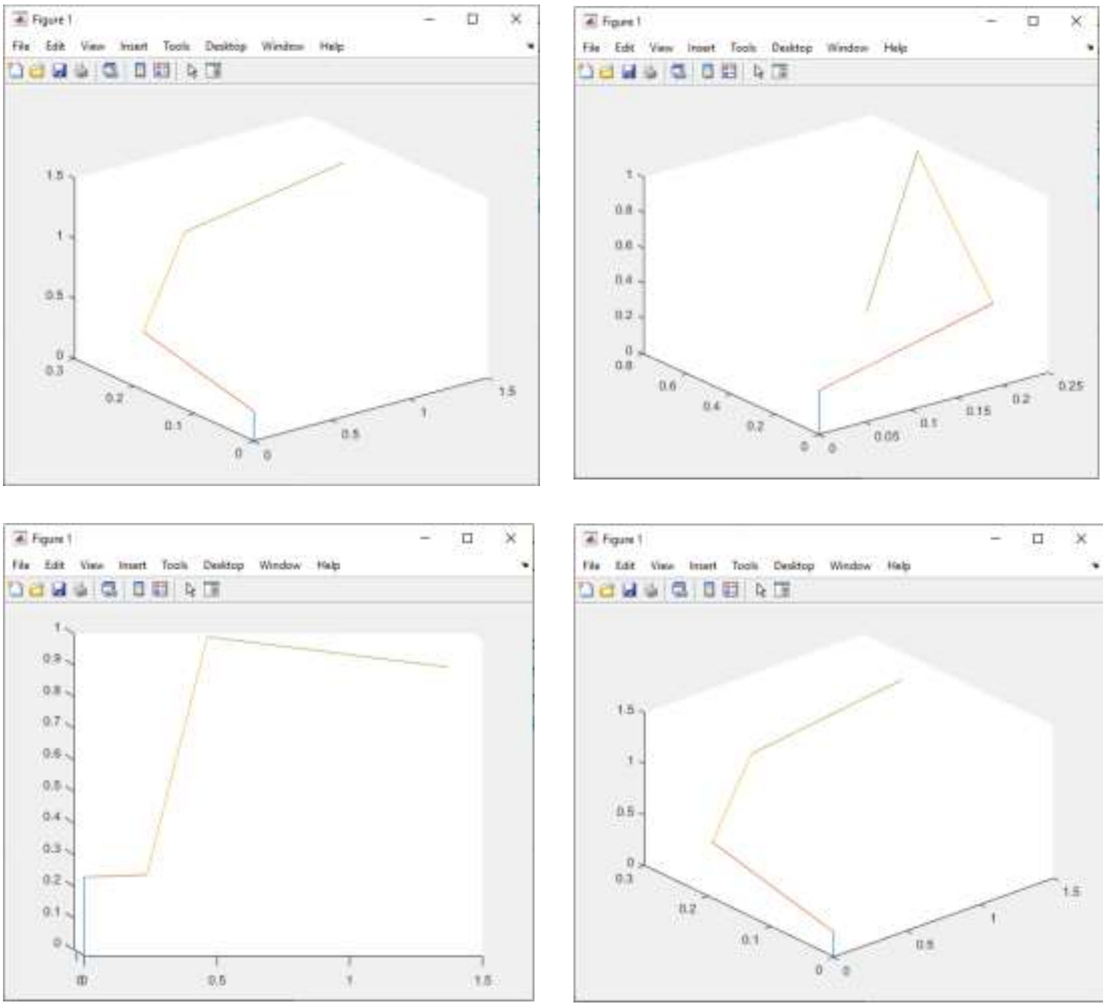
4.2 Denavit-Hartenbergi meetod. Matlab

Autor esitas otsese kinemaatika ülesande lahenduse Denavit-Hartenbergi meetodil Matlab'i koodi kujul (Lisa 3). Samuti võimaldab Matlab programm visualiseerida manipulaatorit ja selle komponentide liikumist [18]. Esimesel koodil autor visuaaliseeris manupulaatori kolmemõõtmelise graafikuna. Selleks autor kasutas funktsiooni plot 3. Tulemust saab vaadata joonisel 4.5.

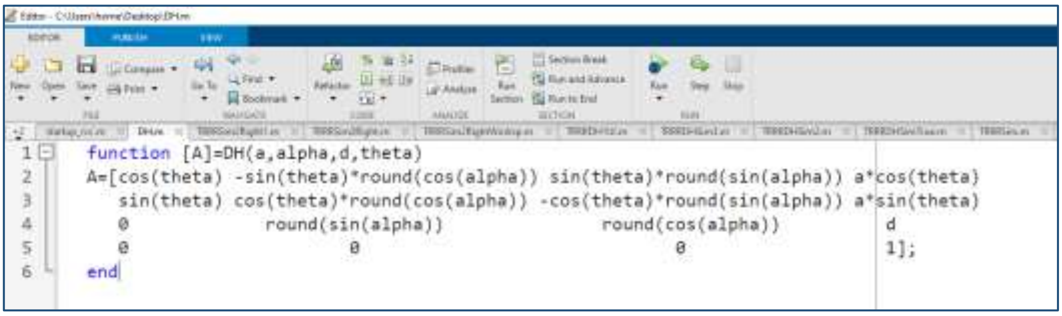
Programm võimaldab ka saada baassüsteemi lõppefektori koordinaatväärtusi.

Programmikoodi tööpõhimõte on järgmine:

- m-faili loomine (vt Joonis 4.6), mis määrab teisendusmaatriksi arvutamise funktsiooni (fail DH.m);
- muutujate määratlemine Denavit-Hartenbergi parameetrite jaoks ja neile arvväärtuste määramine;
- homogeensete teisendusmaatriksite arvutamine DH-faili kirjutatud funktsiooni abil;
- maatriksite järjestikuse korrutamine et saada manipulaatori lõpliku maatriksi;
- iga lüli liikumise esitamine tsükli abil, kus iga järgnev kordus on määratud aja juurdekasvu funktsiooniga. Arvutusvalemite saamiseks manipulaatori liigendite asukoha (x, y, z - koordinaadid) määramiseks baaskoordinaatide süsteemis lõi autor eraldi faili, kus saab teisendusmaatriksi abil koordinaate üldkujul arvutada. Autor kopeeris vastavad valemid ja kandis need põhiprogrammi koodi.



Joonis 4.5. Manipulaatori visualiseerimine



```
function [A]=DH(a,alpha,d,theta)
A=[cos(theta) -sin(theta)*round(cos(alpha)) sin(theta)*round(sin(alpha)) a*cos(theta)
   sin(theta) cos(theta)*round(cos(alpha)) -cos(theta)*round(sin(alpha)) a*sin(theta)
   0          round(sin(alpha))             round(cos(alpha))             d
   0          0                             0                             1];
end
```

Joonis 4.6 m-fail teisendusmaatriksi funktsiooniga (Denavit-Hartenberg'I meetod)

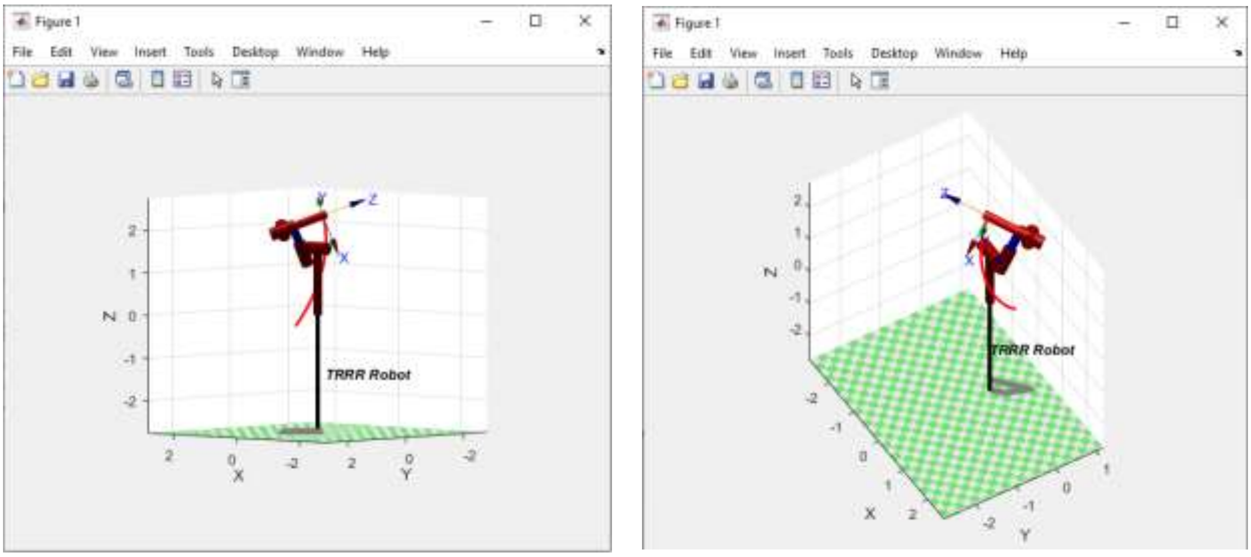
4.3 Denavit-Hartenbergi meetod Robotics Toolbox'i abil

Kasutades Peter Corke'i rakendust Robotics Toolbox, ehitas autor manipulaatori virtuaalse mudeli, simuleerides liikumist mööda etteantud trajektoori [18, 19].

Sel juhul määratakse Denavit-Hartenbergi parameetrid Robotics Toolbox funktsiooni *Link* abil. Numbrid 0 või 1 tähistavad vastavalt pöörlevat või sirgjoonelist liigendit. Manipulaatori liikumise määramiseks kasutas autor *fkine* funktsiooni, mis kuulub ka

Robotics Toolbox’ile. Trajektoori ehitamiseks kasutas autor Robotics Toolbox *jtraj* funktsiooni.

Simulatsiooni tulemus saab näha joonisel 4.7.



Joonis 4.7 Manipulaatori mudel Robotics Toolbox’is

4.4 Simscape kasutamine manipulaatori kinemaatika visualiseerimiseks

Autor ehitas manipulaatori virtuaalse mudeli Simscape keskkonnas, samuti kasutades Robotics Toolbox rakenduse Matlab’is.

Virtuaalsete keskkondade kasutamine võimaldab manipulaatori disaini arvutada teiste geomeetriliste parameetrite abil. Simulatsioon visualiseerib manipulaatori struktuuri, selle liikumist ja juhtimist.

Järjekord:

- Simscape'i kasutamine, et luua robotkäe CAD-mudel
- *Rigid Body Tree* (puitstriktuur) genereerimine konstrueeritud mudeli põhjal
- *Rigid Body Tree* alusel juhtsignaalide lisamine Robotics Toolbox’is

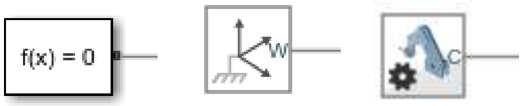
4.4.1 Manipulaatori CAD-mudel

Kasutades programmiteeki koostas autor CAD mudeli.

Simscape programmis mudeli ehitamiseks vajalikud elemendid on kolm juhtelementi (vt Joonis 4.8).

Kolm juhtplokki:

- *Solver* - määrab arvutusvalemi (*Utilities-Solver)*,
- *World Frame* - määrab kogu mehhanismi koordinaatide süsteemi (*Frames and Transforms - World Frame*),
- *Mechanism configuration* - määrab gravitatsiooni ja muud kõigile mehhanismidele mõjuvad jõud (*Utilities- Mechanism configuration)*



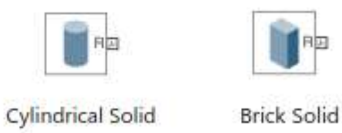
Joonis 4.8 *Solver, World Frame, Mechanism configuration*

Pärast vajaliku kolme elemendi määratlemist autor ehitas manipulaatori mudeli alustades baasist, liikudes järjestikku lõppefektorini.

Esiteks autor lõin puitstruktuuri – *Rigid Body Tree*, mis esindab manipulaatori tegelikku struktuuri. *Rigid Body Tree* koosneb mitmest koosneb komponendist. Struktuuri alusel on baas. Baas määrab maailma koordinaaditiku ja on *Rigid Body* esimene kinnituspunkt [21].

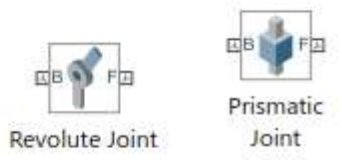
Rigid Body – see on jäiga kerepuu mudeli põhiline ehitusplokk. *Rigid body'de* omavahelisel ühendamisel luuakse vanem-laps tüüpi hierarhia, milles vanemalink on alusküljelt külgnev lüli ja laps on sellele järgnev lisatud lüli. Igal *Rigid body'l* on üks liigend - *Joint*, mis määrab selle *Rigid Body* liikumise selle vanema suhtes. Liigendid (*Joints)* võivad olla fikseeritud, pööratavad ja prismalised [21].

Manipulaatori liigendite ehitamiseks kasutatakse prisma- või silindrikujulisi tahkeid kehasid (Joonis 4.9). Iga tahke keha jaoks saate määrata geomeetrilised mõõtmed.



Joonis 4.9 Ehitusplokid

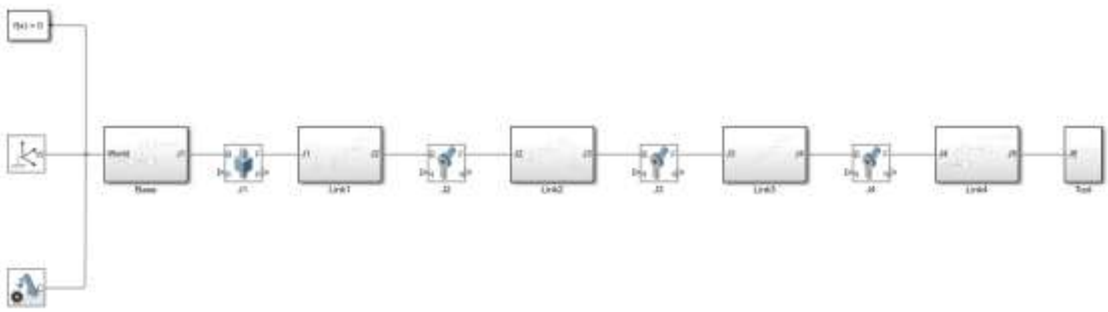
Koordinaatsüsteemide omavaheliseks ühendamiseks kasutatakse Rigid Transform elementi. Mudeli koostamisel ja koordinaatide telgede asukohtade reguleerimisel Simscape'is tuleks arvestada mõningate reeglitega. Pöördligend *(revolute joint)* puhul määrab programm pöördeteljeks z-telje. Translatsioonilink *(Prismatic joint)* liigub piki z-telge (vt Joonis 4.10).



Joonis 4.10 Liigendid *(Joints)*

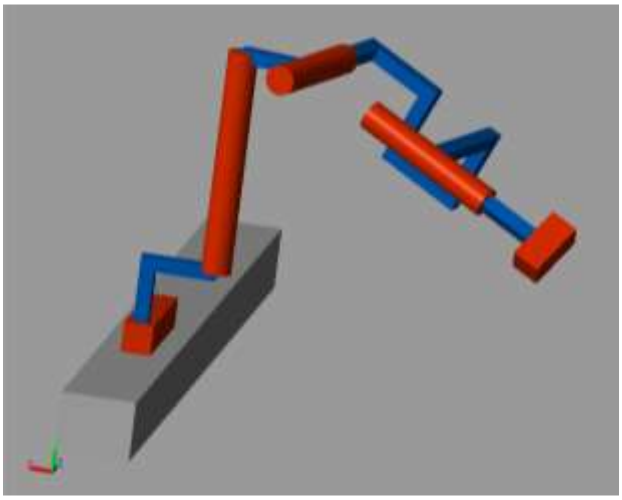
Seega ehitas autor järjestikku kogu manipulaatori struktuuri (Vt Joonis 4.11) [22]. Autor konfigureeris manipulaatori iga liigendi parameetrid vastavalt algandmetele.

Iga liigendiga seotud elemendid ühendatakse vastavatesse rühmadesse. Iga rühma koosseis on toodud joonistel 4.13, 4.14, 4.15, 4.16, 4.17, 4.18.

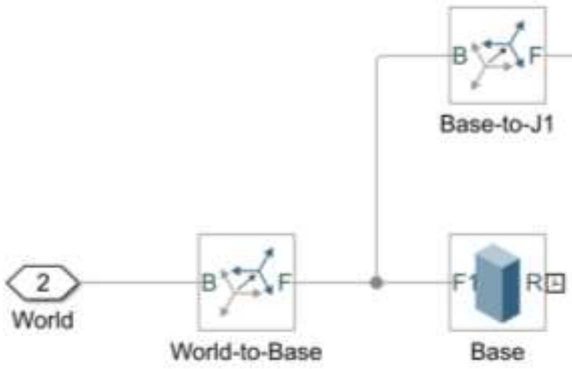


Joonis 4.11 *Rigid Body Tree*

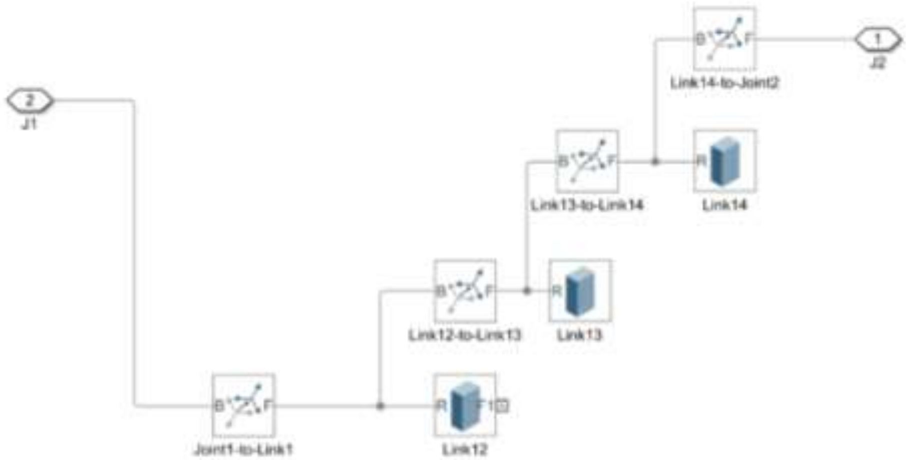
Nüüd saab manipulaatori mudelit kasutada testimiseks ja simulatsioonide loomiseks. Pärast manipulaatori struktuuri loomist saab käivitada simulatsiooni režiimi ja vaadata selle mudelit (vt Joonis 4.12).



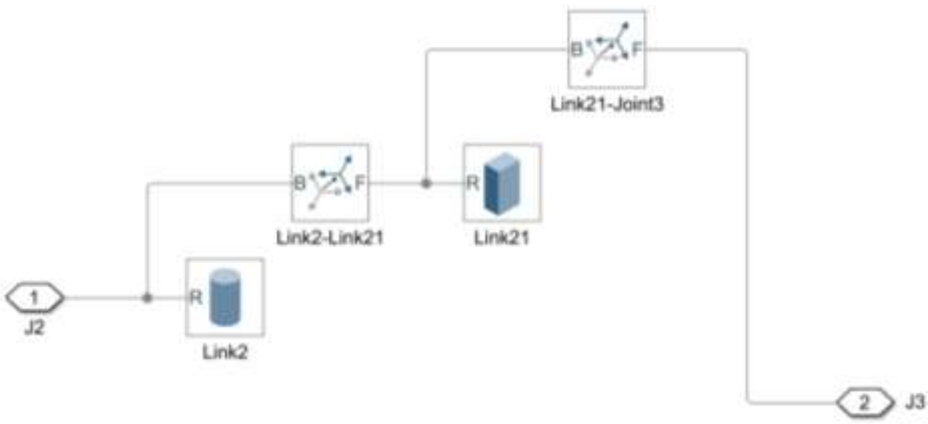
Joonis 4.12 Manipulaatori struktuur



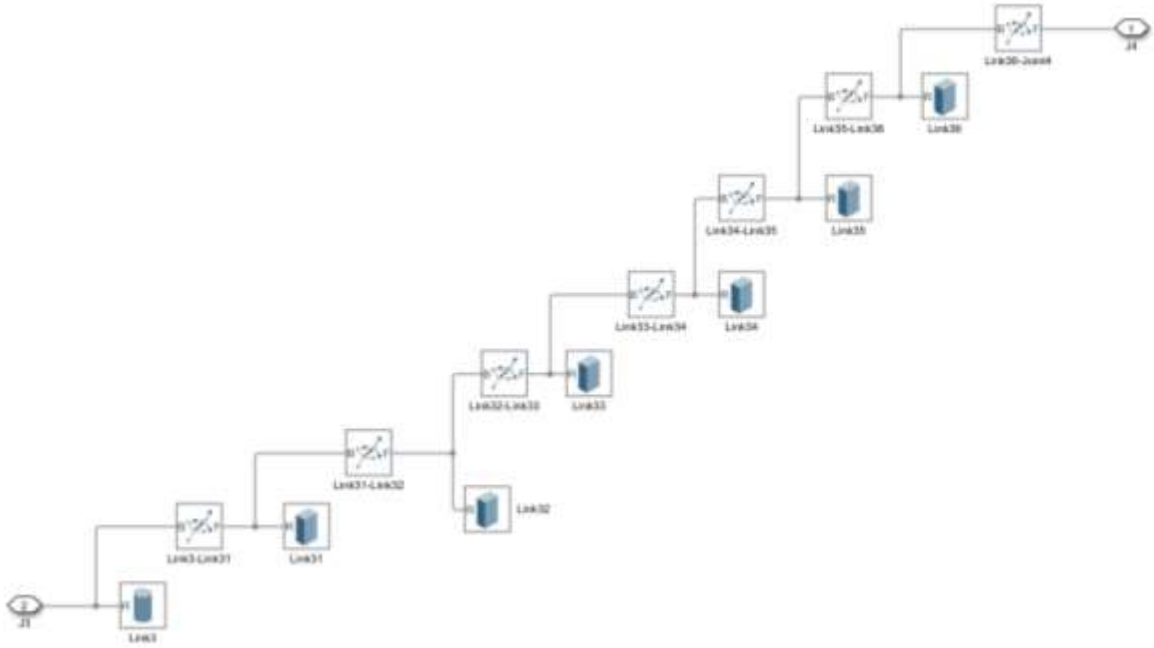
Joonis 4.13 Baasliigend *(Base)*



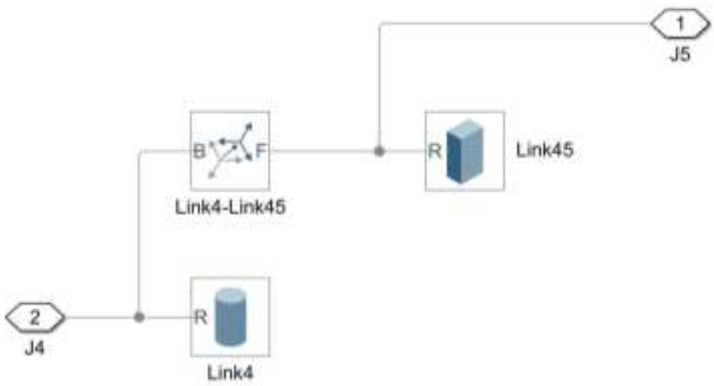
Joonis 4.14 Liigend 1 *(Joint 1)*



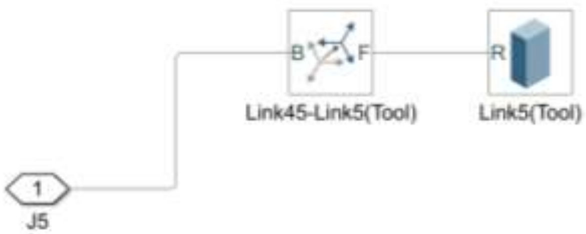
Joonis 4.15 Liigend 2 *(Joint 2)*



Joonis 4.16 Liigend 3 *(Joint 3)*

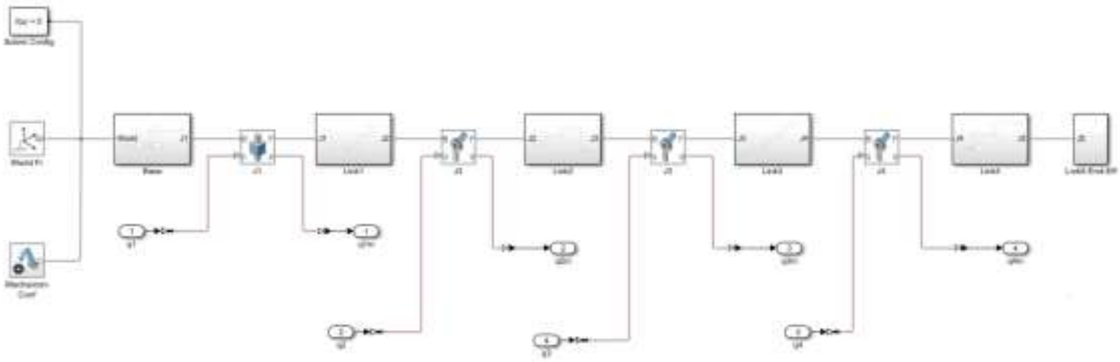


Joonis 4.17 Liigend 4 *(Joint 4)*



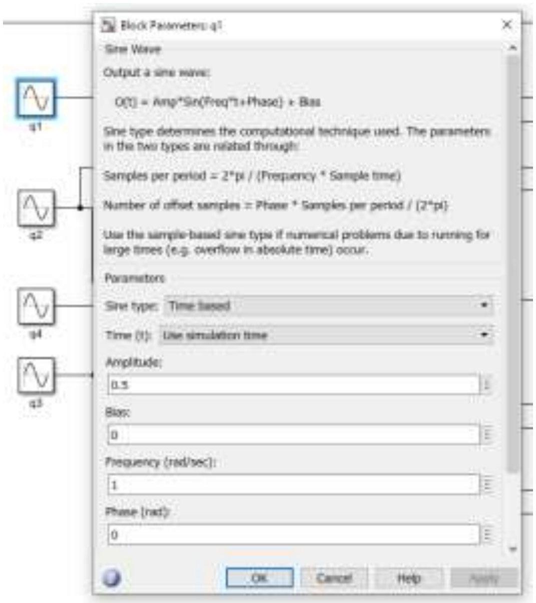
Joonis 4.18 Liigend 5 *(Joint 5)*

Simulink'i süsteemi ühendamiseks füüsilise süsteemiga kasutame konvertereid, mis on programmiteegis. Ühenduse sisendisse paigaldatakse muundurid Simulink'ist füüsilisse süsteemi, väljundisse füüsilisest süsteemist Simulinki. Linkide sisendparameetrid on pöördenurgad (vt Joonis 4.19). Autor ühendas kogu manipulaatori struktuuri ühte rühma, mida nimetas *Angles.*



Joonis 4.19 Struktuur konverteritega

Nurga väärtuste määramiseks kasutatakse siinuslaine generaatorit (vt Joonis 4.20).



Joonis 4.20 Sinuslaine generaatori seadistused

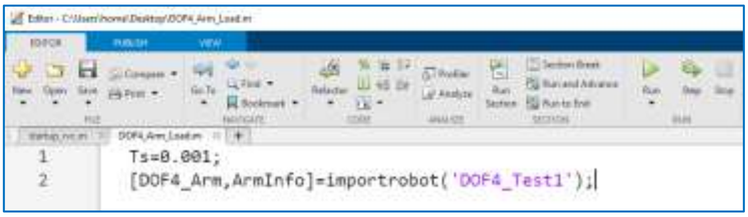
4.4.2 Robotics Toolbox'i ühendamine

Järgmine samm otsese kinemaatika ülesande lahendamiseks on Robotics Toolbox'i ühendamine. Valime *Get Transform* plokk *(Manipulator Algorythm – Get Transform).*

Get Transform plokk nõuab *Rigid Body Tree* ja konfiguratsiooni, mis sisaldab koordinaatsüsteemide asukohta ja nende orientatsiooni.

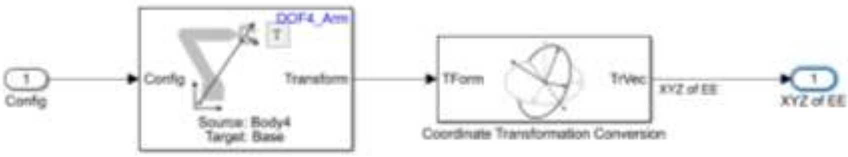
Robotics Toolbox'is manipulaatoriga töötamise jätkamiseks peab looma manipulaatori puu struktuuri – *Rigid Body Tree.*

Rigid Body Tree määramiseks peab looma ja käivitama väikese m.-faili, mis sisaldab impordifunktsiooni (vt Joonis 4.21).

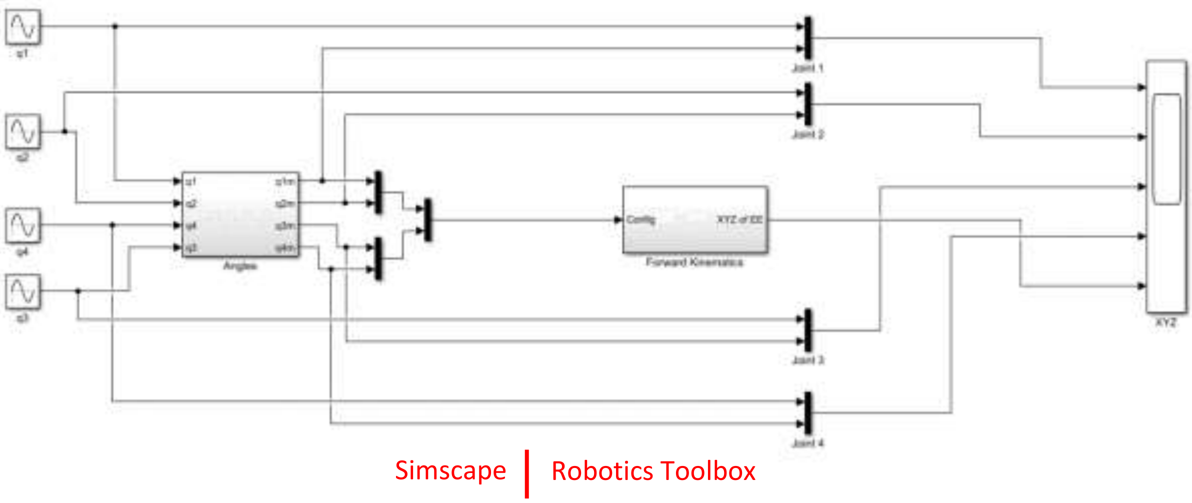


Joonis 4.21 *Importrobot* funktsiooni käivitamine

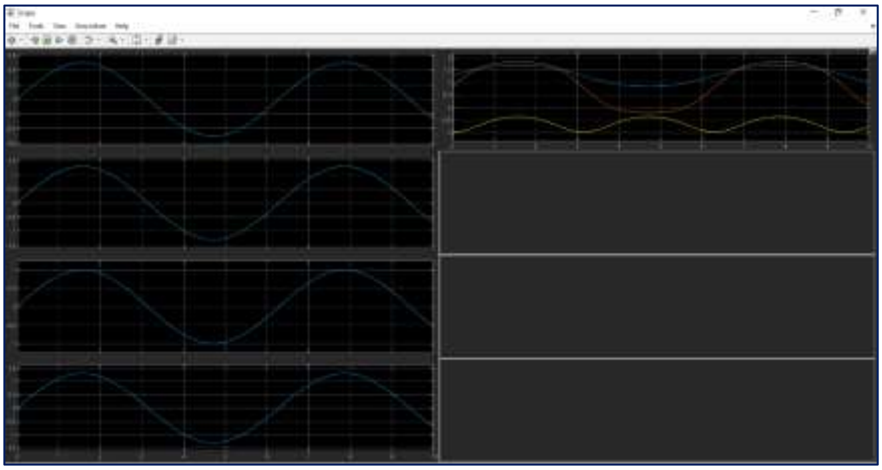
Positsiooniväärtuste saamiseks homogeensest teisendusmaatriksist kasutas autor *Coordinate Transformation Conversion* plokk (vt Joonis 4.22). Plokk on konfigureeritud nii, et väljundiks on lõppefektori asukoht. Koordinaatide väärtused kuvatakse sellises plokis nagu *Scope* (vt Joonis 4.23, 4.24).



Joonis 4.22 Forward kinematics



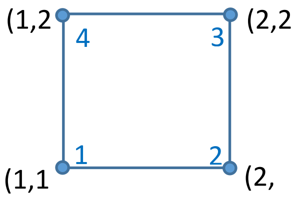
Joonis 4.23 Euleri nurkade mõõtmine



Joonis 4.24 Scope

4.5 Manipulaatori liikumise visualiseerimine mööda trajektoori

Järgmises ülesandes testis autor Simscape'i võimeid pöördkinemaatika ülesande lahendamisel [23]. Sel juhul määrati manipulaatorile teatud liikumistrajektoor (vt Joonis 4.25). Autor võttis yz tasapinnas asuva ristküliku koordinaatidega [(1,1); (1,2); (2,2); (2,1)].



Joonis 4.25 Trajektoor

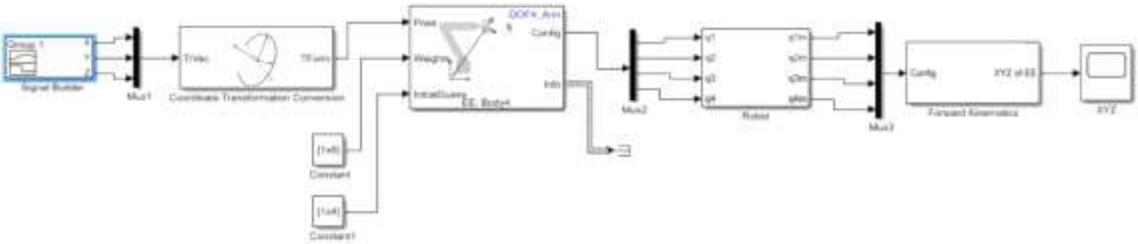
Manipulaatori lõppefektor liigub järjestikku punktidesse 1,2,3,4.

Ülesande täitmiseks kasutas autor otsese- ja pöördkinemaatika plokke (vt Joonis 4.26). Signaali teisendamiseks teatud liigendinurkadeks peab kasutama *Inverse Kinematics* plokki. Üks ploki sisendparameetritest on poos – lõppefektori asukoht. Translatsioonivektori homogeenseks teisendusmaatriksiks teisendamiseks on vaja *Coordinate Transformation* plokk [23].

Muud ploki sisendparameetrid on kaal ja lähteasend. Sel juhul määras autor igale kolmele teljele sama kaalu. Algpositsiooni jaoks kasutas autor nullkoordinaate [23].

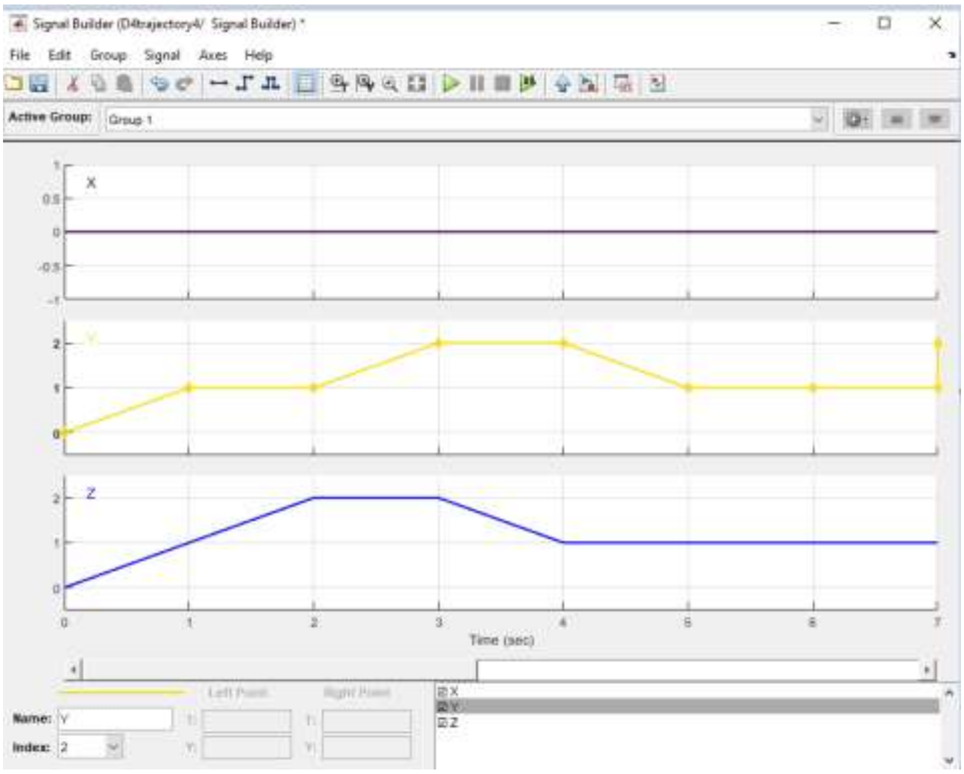
Autor rakendas Inverse Kinematics plokile sama *Rigid Body Tree*, mis oli varem tehtud *Forward kinematics* jaoks (m-failist DOF4_load).

Lõppefektori asukoha väärtust saab kasutada tagasisidena ja edastada süsteemi sisendisse. Sel juhul analüüsitakse alg- ja lõppväärtuste erinevust. Kui näitudes on erinevusi, peate installima täiendava juhtimissüsteemi. Kuid selles töös ei võtnud autor ette nii sügavat analüüsi.



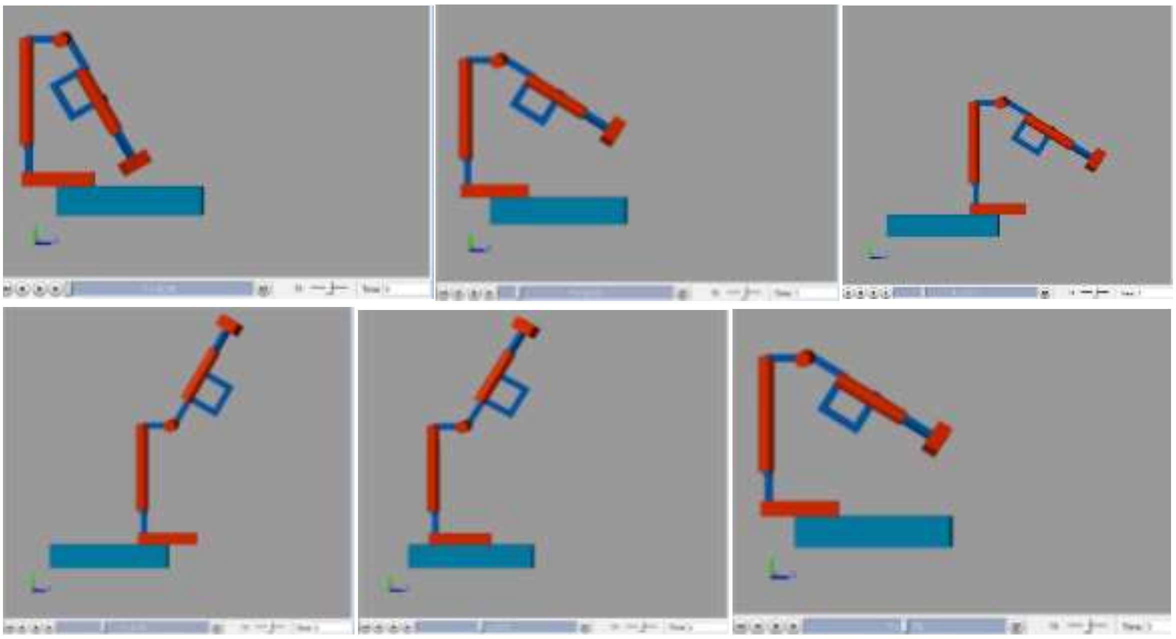
Joonis 4.26. Otsese ja pöördkinematika plokid

Trajektoori loomiseks kasutas autor *Signal Builder.* Selle seadetes saab määrata manipulaatori lõppefektori liikumise koordinaadid x,y,z tasapinnal (vt Joonis 4.27). Sel juhul määras autor koordinaatideks 5 punkti, alustades manipulaatori neutraalasendist - punktist koordinaatidega (0,0). Trajektooripunktide vahel liikumine võtab aega ühe sekundi. Antud trajektoori puhul ei toimu liikumist x-tasandil.



Joonis 4.27 *Signal builder*

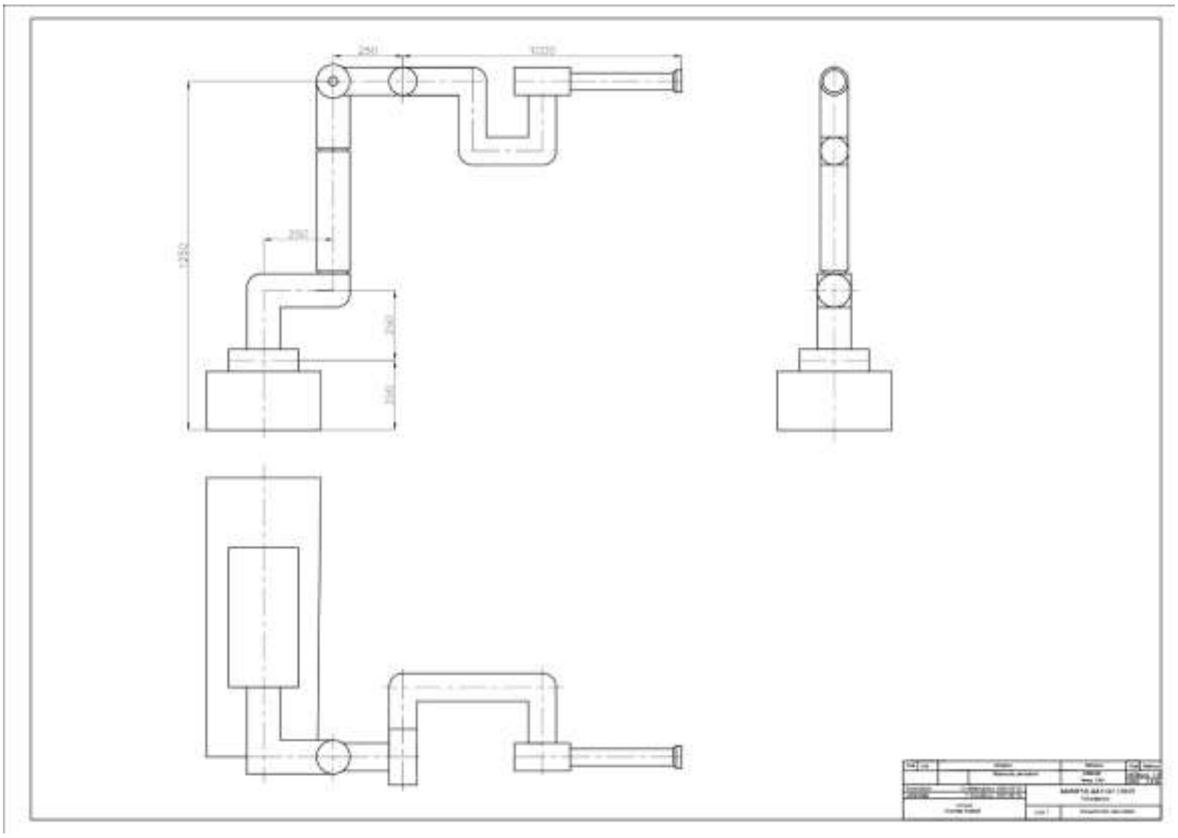
Autor kasutas saadud struktuuri liikumise simuleerimiseks ja visualiseerimiseks (vt Joonis 4.28, Lisa 8).



Joonis 4.28 Manipulaatori positsioon

5. GRAAFILINE OSA

Töö graafilise osana esitas autor manipulaatori skemaatilise joonise koos esialgsete geomeetriliste parameetritega (vt Joonis 5.1, Lisa 9). Selle lõputöö keskmes on manipulaatori matemaatiline mudel. Täpsed projekteerimis- ja tehnoloogilised lahendused ning konstruktsioonid on jäetud väljapoole lõputöö käsitlust.



Joonis 5.1. Manipulaatori mõõtmed ja konstruktsioon

KOKKUVÕTE

Lõputöö käigus uuris autor nelja vabadusastmega manipulaatori TRRR struktuuri ja matemaatilist mudelit.

Autor lahendas otsese kinemaatika ülesande kahe meetodi abil: otsemeetodi (rotatsioonimaatriksi meetodi) ja Denavit-Hartenbergi meetodi abil. Autor esitas arvutused nii Excel'i faili, Matlab'i koodi kui ka Simscape programmi abil teostatud simulatsioonifailide näol. Kõnealuse manipulaatori ülesehituse paremaks mõistmiseks lisas autor lõputööle ka manipulaatori joonise ja tööfailid. Autor peab võimalikuks kasutada neid faile, et koolitada insenere konkreetsel manipulaatori näitel otsese kinemaatika ülesande lahendamisi meetodeid. Autor teeb ettepaneku kasutada faile ja meetodi teoreetilist kirjeldust erinevate parameetrite mõju demonstreerimiseks, manipulaatori geomeetrilise mudeli, prototüüpide ehitamiseks ja visualiseerimiseks, erinevate lähteandmete testimiseks ning ka arvutuste tulemuste enesekontrolliks.

Lõputöös autor käsitles näitena kasutati TRRR tüüpi manipulaatorit (ühe translatsiooni- ja kolme pöördliigendiga manipulaator). Autori sõnul võimaldab seda tüüpi manipulaatori kasutamine selgemalt demonstreerida meetodeid otsese kinemaatika ülesande lahendamiseks. Lõputöö kirjutamise ajal suurem osa avalikult leitavatest teabe- ja koolitusmaterjalidest oli pühendatud rotatsiooniliigenditega manipulaatoritele (RRR ja RRRR tüübid). Autor usub, et õppimiseks sobib paremini nii sirgjoonelise- kui ka pöördliideseid sisaldava manipulaatori konstruktsiooni kasutamine näitena. Sel juhul võetakse ülesannete lahendamisel arvesse nii sirgjoonelise- kui ka rotatsiooni kinemaatiliste paaride iseärasusi.

Lõputöö valmimise käigus avanes autoril võimalus võrrelda kahte meetodit otsese kinemaatika probleemi lahendamiseks (rotatsioonimaatriksi meetod ja Denavit-Hartebergi meetod). Põhiparameetreid kasutav manipulaatori mudeli kirjeldus on töö autori sõnul üsna universaalne. Meetodi tulemusi on mugav kasutada edasisteks arvutusteks, näiteks Euleri nurkade arvutamiseks. Denavit-Hartenbergi parameetreid saab kasutada töö virtuaalse mudeli ja sellega seotud simulatsioonide loomisel, näiteks Robotics Toolbox rakenduses. Meetodi keerukus võib autori arvates seisneda manipulaatori liigenditega seotud koordinaattelgede õige asukoha määramises, võttes arvesse meetodi reegleid ja piiranguid.

Ülesande lahendamine Denavit-Hartenbergi meetodil üldiselt võib muutujate rohkuse tõttu samuti raskusi tekitada. Käesolevas töös on autor tuletanud ja demonstreerinud valemit konkreetse TRRR-manipulaatori arvutamiseks. Seda valemit kasutatakse Excel'i failivormingus esitatud metodiliste materjalide aluseks. Tulevaste projekteerijate jaoks on autori arvates otstarbekam kasutada mõlemat meetodit kasutades kinemaatikaülesannete lahendamisel vastavaid digitaalseid keskkondi või lahendada ülesanne kohe konkreetsete arvväärtustega. See lihtsustab arvutusi oluliselt.

Manipulaatori disaini visualiseerimiseks, prototüübi ehitamiseks ja simulatsioonide läbiviimiseks kasutas autor Matlab'i keskkonda ja selle rakendusi, nagu Robotics Toolbox ja Simscape. Nendel digitaalsetel keskkondadel on lai funktsionaalsus, need võimaldavad teil luua nii otse programmis manipulaatori mudeli kui ka integreerida teistes rakendustes, näiteks SolidWorks'is, loodud 3D-mudeleid. Programm sisaldab palju tööriistu, mis võimaldavad määrata manipulaatorile juhtimist ja teha vajalikke projekteerimisarvutusi, nagu pöördkinemaatika ülesande lahendamine, otsese kinemaatika ülesande lahendamine, otse- ja pöörddünaamika ülesannete lahendamine. Käesolevas töös ei kasutanud autor simulatsioonide loomisel kõiki programmis olemasolevaid funktsioone. Samuti ei ole intergreerinud autor 3D-mudelit Matlab'i keskkonnasse. Selles töös valis autor mudeli ehitamise meetodi otse Simscape'is. Töö põhieesmärk on näidata mitmeid meetodeid otsese kinemaatilise probleemi lahendamiseks konkreetse manipulaatori põhjal, autori meeles on täidetud. Samuti lõi autor abistavad metoodilised materjalid.

Autor peab võimalikuks edaspidistes töödes jätkata ka teisi lähenemisi TRRR manipulaatori simulatsiooni loomisele, edasisele ehitamisele ja uurimisele.

SUMMARY

Industrial robots, due to their high precision, flexibility and productivity, are widely used in various fields of production. Understanding the typical structure of a manipulator, the mathematical model that underlies it is necessary for future engineers for design, calculation, selection of drives and other engineering operations.

Designing a robotic manipulator consists of several stages, one of which is solving kinematics problems—direct and inverse.

In this work, the author examined the solution of the direct kinematics problem using two methods based on a TRRR manipulator which has one translational and three rotational joints. The author considers the choice of this type of manipulator to be a good solution for compiling training materials for future engineers, in contrast to the most common RRR type. The TRRR manipulator allows you to study the structure and consider the solution of design problems, taking into account the features of both rotational and translational joints.

The author used the direct method (the rotation matrix method) and the Denavit-Hartenberg method to solve the direct kinematics problem. For each method, the author described the sequence of actions necessary to solve the problem, showed the mathematical calculation step by step, and also attached additional materials created for calculating the manipulator. Comparing the methods with each other, the author concluded that the Denavit-Hartenberg method is more universal. After determining the set of Denavit-Hartenberg parameters, one can proceed with the calculation and study of the manipulator using various digital environments such as Excel, Matlab (with Simscape and Robotic Toolbox), and SolidWorks applications for design, calculations, and simulation.

The result of this work is a mathematical calculation of the working model, as well as auxiliary teaching materials created in digital environments. The author did not set out to study the dynamics of the manipulator, leaving this as a goal for his future works.

KASUTATUD KIRJANDUSE LOETELU

1. Gurjeet Singh. V.K. Banga „Robots and its types for industrial applications”. [*Online*] https://www.sciencedirect.com/science/article/pii/S2214785321081682 (28.01.2024).

2. К.В. Фролов. Е.И. Воробьев «Механика промышленных роботов». Москва. Высшая школа. 1988.

3. Ravi Rao. “7 Types of Industrial Robots: Advantages, Disadvantages, Applications, and More”. [*Online*] https://www.wevolver.com/article/7-types-of-industrial-robots-advantages-disadvantages-applications-and-more (03.03.2020).

4. International ISO standart. iTeh Standart preview [*Online*] https://cdn.standards.iteh.ai/samples/75539/1bc8409322eb4922bf680e15901852d2/ISO-8373-2021.pdf (28.01.2024).

5. International ISO standart ISO 10360- 12:2016 [*Online*] https://www.iso.org/standard/63931.html (28.01.2024).

6. Fanuc [*Online*] https://www.fanuc.eu/uk/en/robots/robot-filter-page/lrmate-series/lrmate-200-id (28.01.2024).

7. Mitsubishi [*Online*] https://www.mitsubishielectric.com/fa/products/rbt/robot/pmerit/vertical/crseries/index.html (28.01.2024).

8. Yamaha [*Online*] https://www.yamaha-usa-robotics.com/robotics/data-sheets/YA-R3F.pdf (28.01.2024).

9. Vertically articulated robots YA series [*Online*] https://www.youtube.com/watch?v=wNY01XEi_nI&t=81s (28.01.2024).

10. Jorge Angeles „Fundamentals of Robotic Mechanical Systems. Theory, methods and algorithms‟. Third edition. Springer Science+Business Media, LLC, 2007

11. Arumalla Johnson, M.Venkatesh, Department of Mechanical Engineering, Kamala Institute of Technology and Science, “Forward and inverse kinematic analysis of 4-DOF TRRR robotic arm using matlab” , Journal of emerging tehnologies and innovative research (Jetir), vol 8 (28.01.2020).

12. Tallinna Tehnikaülikool. Elektriajamite ja jõuelektroonika instituut. Tõnu Lehtla. „Robotitehnika‟. Tallinn, 2006.

13. О.И. Борисов, В.С. Громов, А.А. Пыркин „МЕТОДЫ УПРАВЛЕНИЯ РОБОТОТЕХНИЧЕСКИМИ ПРИЛОЖЕНИЯМИ‟. Учебное пособие. Университет ИТМО. 2016.

14. Didier Pascault „Articulated Mechanical Structures. An introduction. Part 1". UPJV-Amiens. 2022.

15. Kevin M. Lynch and Frank C. Park „Modern robotics mechanics, planning, and control‟. Cambridge University Press, 2017.

16. Peter Corke „Robotics, vision and control. Fundamental algorithms in matlab‟. Springer International Publishing AG 2017.

17. ForwardKinematics of Robot Arm in MATLAB [*Online*] https://www.youtube.com/watch?v=gXprzY8UcTU&t=2328s (05.04.2024).

18. Matlab tutorial on Forward Kinematics Visualization [*Online*] https://www.youtube.com/watch?v=gCDyvPN9lrE&t=1718s (05.04.2024).

19. How to define a robot with the Denavit and Hartenberg parameters [*Online*] https://www.youtube.com/watch?v=bnShNV0D1ds (01.05.2024).

20. Forward Kinematics 3 DOF Robot Arm in 3D [*Online*] https://www.youtube.com/watch?v=bgiwFDkrzmw (28.01.2024).

21. Rigid Body Tree Robot Model [*Online*] https://www.mathworks.com/help/robotics/ug/rigid-body-tree-robot-model.html (10.05.2024)

22. How to design Robots using MATLAB 2021 | SimScape Toolbox Robotics System Toolbox [*Online*] https://www.youtube.com/watch?v=_8YCc3pJDPI&t=2199s (28.01.2024).

23. Drive Robot Using Inverse Kinematics in Simulink | MATLAB 2021 Robotic System Toolboox [*Online*] https://www.youtube.com/watch?v=20ENJ7jOJzE (16.02.2024).

LISA 1. TEISENDUSE MAATRIKSITE ARVUTAMINE

M_0^1 maatriks

$$M_0^1 = \begin{bmatrix} cos\theta_1 & -sin\theta_1 cos\alpha_1 & sin\theta_1 sin\alpha_1 & a_1 cos\theta_1 \\ sin\theta_1 & cos\theta_1 cos\alpha_1 & -cos\theta_1 sin\alpha_1 & a_1 sin\theta_1 \\ 0 & sin\alpha_1 & cos\alpha_1 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} cos90° & -sin90°\cos(90°) & sin90°\sin(90°) & 0 \cdot cos90° \\ sin90° & cos90°\cos(90°) & -cos90°\sin(90°) & 0 \cdot sin90° \\ 0 & \sin(90°) & \cos(90°) & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0{,}25 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

M_1^2 maatriks

$$M_1^2 = \begin{bmatrix} cos\theta_2 & -sin\theta_2 cos\alpha_2 & sin\theta_2 sin\alpha_2 & a_2 cos\theta_2 \\ sin\theta_2 & cos\theta_2 cos\alpha_2 & -cos\theta_2 sin\alpha_2 & a_2 sin\theta_2 \\ 0 & sin\alpha_1 & cos\alpha_1 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} cos0° & -sin0°\cos(-90°) & sin0°\sin(-90°) & 0{,}25cos0° \\ sin0° & cos0°\cos(-90°) & -cos0°\sin(-90°) & 0{,}25sin0° \\ 0 & \sin(-90°) & \cos(-90°) & 0{,}25 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0{,}25 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0{,}25 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

M_2^3 maatriks

$$M_2^3 = \begin{bmatrix} cos\theta_3 & -sin\theta_3 cos\alpha_3 & sin\theta_3 sin\alpha_3 & a_3 cos\theta_3 \\ sin\theta_3 & cos\theta_3 cos\alpha_3 & -cos\theta_3 sin\alpha_3 & a_3 sin\theta_3 \\ 0 & sin\alpha_3 & cos\alpha_3 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} \cos(-45°) & -\sin(-45°)\cos(-90°) & \sin(-45°)\sin(-90°) & 0{,}25 \cdot \cos(-45°) \\ \sin(-45°) & \cos(-45°)\cos(-90°) & -\cos(-45°)\sin(-90°) & 0{,}25 \cdot \sin(-45°) \\ 0 & \sin(-90°) & \cos(-90°) & 0{,}75 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} \frac{\sqrt{2}}{2} & 0 & \frac{\sqrt{2}}{2} & 0,25 \cdot \frac{\sqrt{2}}{2} \\ -\frac{\sqrt{2}}{2} & 0 & \frac{\sqrt{2}}{2} & -0,25 \cdot \frac{\sqrt{2}}{2} \\ 0 & -1 & 0 & 0.75 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$\boldsymbol{M_3^4}$ **maatriks**

$$M_3^4 = \begin{bmatrix} cos\theta_4 & -sin\theta_4 cos\alpha_4 & sin\theta_4 sin\alpha_4 & a_4 cos\theta_4 \\ sin\theta_4 & cos\theta_4 cos\alpha_4 & -cos\theta_4 sin\alpha_4 & a_4 sin\theta_4 \\ 0 & sin\alpha_4 & cos\alpha_4 & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} \cos(-60°) & -\sin(-60°)\cos(-90°) & \sin(-60°)\sin(-90°) & 0 \cdot \cos(-60°) \\ \sin(-60°) & \cos(-60°)\cos(-90°) & -\cos(-60°)\sin(-90°) & 0 \cdot \sin(-60°) \\ 0 & \sin(-90°) & \cos(-90°) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} \frac{1}{2} & 0 & \frac{\sqrt{3}}{2} & 0 \\ -\frac{\sqrt{3}}{2} & 0 & \frac{1}{2} & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$\boldsymbol{M_4^5}$ **maatriks**

$$M_4^5 = \begin{bmatrix} cos\theta_5 & -sin\theta_5 cos\alpha_5 & sin\theta_5 sin\alpha_5 & a_5 cos\theta_5 \\ sin\theta_5 & cos\theta_5 cos\alpha_5 & -cos\theta_5 sin\alpha_5 & a_5 sin\theta_5 \\ 0 & sin\alpha_5 & cos\alpha_5 & d_5 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} cos0° & -sin0° cos0° & sin0° sin0° & 0 \cdot cos0° \\ sin0° & cos0° cos0° & -cos0° sin0° & 0 \cdot sin0° \\ 0 & sin0° & cos0° & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

Leiame maatriksite korrutamise teel (valem 2) viimase tööriista haardega seotud koordinaatsüsteemi asukoha baaskoordinaadisüsteemis. Korrutage järjestikku.

$M_0^5 = M_0^1 M_1^2 M_2^3 M_3^4 M_4^5$ (2)

$\boldsymbol{M_0^2}$ **maatriks**

$$M_0^2 = M_0^1 M_1^2 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0{,}25 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0{,}25 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0{,}25 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 & 0{,}25 \\ 1 & 0 & 0 & 0{,}25 \\ 0 & 0 & 1 & 0{,}25 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

M_0^3 maatriks

$$M_0^3 = M_0^1 M_1^2 M_2^3 = \begin{bmatrix} 0 & -1 & 0 & 0{,}25 \\ 1 & 0 & 0 & 0{,}25 \\ 0 & 0 & 1 & 0{,}25 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{\sqrt{2}}{2} & 0 & \frac{\sqrt{2}}{2} & 0{,}25 \cdot \frac{\sqrt{2}}{2} \\ -\frac{\sqrt{2}}{2} & 0 & \frac{\sqrt{2}}{2} & -0{,}25 \cdot \frac{\sqrt{2}}{2} \\ 0 & -1 & 0 & 0.75 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} \frac{\sqrt{2}}{2} & 0 & -\frac{\sqrt{2}}{2} & 0{,}25 \cdot \frac{\sqrt{2}}{2} + 0{,}25 \\ \frac{\sqrt{2}}{2} & 0 & \frac{\sqrt{2}}{2} & 0{,}25 \cdot \frac{\sqrt{2}}{2} + 0{,}25 \\ 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

M_0^4 maatriks

$$M_0^4 = M_0^1 M_1^2 M_2^3 M_3^4 = \begin{bmatrix} \frac{\sqrt{2}}{2} & 0 & -\frac{\sqrt{2}}{2} & 0{,}25 \cdot \frac{\sqrt{2}}{2} + 0{,}25 \\ \frac{\sqrt{2}}{2} & 0 & \frac{\sqrt{2}}{2} & 0{,}25 \cdot \frac{\sqrt{2}}{2} + 0{,}25 \\ 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{1}{2} & 0 & \frac{\sqrt{3}}{2} & 0 \\ -\frac{\sqrt{3}}{2} & 0 & \frac{1}{2} & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} \frac{\sqrt{2}}{4} & \frac{\sqrt{2}}{2} & \frac{\sqrt{2}\sqrt{3}}{4} & 0{,}25 \cdot \frac{\sqrt{2}}{2} + 0{,}25 \\ \frac{\sqrt{2}}{4} & -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}\sqrt{3}}{4} & 0{,}25 \cdot \frac{\sqrt{2}}{2} + 0{,}25 \\ \frac{\sqrt{3}}{2} & 0 & -\frac{1}{2} & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

M_0^5 maatriks

$$M_0^5 = M_0^1 M_1^2 M_2^3 M_3^4 M_4^5 = \begin{bmatrix} \frac{\sqrt{2}}{4} & \frac{\sqrt{2}}{2} & \frac{\sqrt{2}\sqrt{3}}{4} & 0{,}25 \cdot \frac{\sqrt{2}}{2} + 0{,}25 \\ \frac{\sqrt{2}}{4} & -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}\sqrt{3}}{4} & 0{,}25 \cdot \frac{\sqrt{2}}{2} + 0{,}25 \\ \frac{\sqrt{3}}{2} & 0 & -\frac{1}{2} & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$=\begin{bmatrix} \frac{\sqrt{2}}{4} & \frac{\sqrt{2}}{2} & \frac{\sqrt{2}\sqrt{3}}{4} & \frac{\sqrt{2}\sqrt{3}}{4}+0,25\cdot\frac{\sqrt{2}}{2}+0,25 \\ \frac{\sqrt{2}}{4} & -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}\sqrt{3}}{4} & \frac{\sqrt{2}\sqrt{3}}{4}+0,25\cdot\frac{\sqrt{2}}{2}+0,25 \\ \frac{\sqrt{3}}{2} & 0 & -\frac{1}{2} & -\frac{1}{2}+1 \\ 0 & 0 & 0 & 1 \end{bmatrix}=$$

$$=\begin{bmatrix} \frac{\sqrt{2}}{4} & \frac{\sqrt{2}}{2} & \frac{\sqrt{2}\sqrt{3}}{4} & \frac{\sqrt{2}\sqrt{3}}{4}+0,25\cdot\left(\frac{\sqrt{2}}{2}+1\right) \\ \frac{\sqrt{2}}{4} & -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}\sqrt{3}}{4} & \frac{\sqrt{2}\sqrt{3}}{4}+0,25\cdot\left(\frac{\sqrt{2}}{2}+1\right) \\ \frac{\sqrt{3}}{2} & 0 & -\frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\overrightarrow{OP}_{/Ro}=\begin{bmatrix} \frac{\sqrt{2}\sqrt{3}}{4}+0,25\cdot\left(\frac{\sqrt{2}}{2}+1\right) \\ \frac{\sqrt{2}\sqrt{3}}{4}+0,25\cdot\left(\frac{\sqrt{2}}{2}+1\right) \\ \frac{1}{2} \end{bmatrix}\approx\begin{bmatrix} 1,039 \\ 1,039 \\ 0,5 \end{bmatrix}$$

LISA 2. ROTATSIOONIMAATRIKSI MEETOD. KOOD

```
clc;
clear all;
close all;
syms theta
Xrot=[1 0 0; 0 cos(theta) -sin(theta); 0 sin(theta) cos(theta)];
Yrot=[cos(theta) 0 sin(theta); 0 1 0; -sin(theta) 0 cos(theta)];
Zrot=[cos(theta) -sin(theta) 0; sin(theta) cos(theta) 0; 0 0 1];
syms theta1 theta2 theta3 theta4 theta5 tr1 l1z l2y l2z l3 l4 l5
OO1=[tr1; 0; l1z];
O1O2=[0; l2y; l2z];
O2O3=[0; l3; 0];
O3O4=[0; l4; 0];
O4P=[0; l5; 0];
theta=theta1;
theta1=0;
%theta=eval('theta');
theta1=eval('theta1');
%M01=Xrot;
M01=[1 0 0; 0 cos(theta1) -sin(theta1); 0 sin(theta1) cos(theta1)];
theta=theta2;
theta2=-pi/4;
theta2=eval('theta2');
%M12=Zrot;
M12=[cos(theta2) -sin(theta2) 0; sin(theta2) cos(theta2) 0; 0 0 1];
theta=theta3;
theta3=-pi/6;
theta3=eval('theta3');
%M23=Xrot;
M23=[1 0 0; 0 cos(theta3) -sin(theta3); 0 sin(theta3) cos(theta3)];
theta=theta4;
theta4=0;
theta4=eval('theta4');
%M34=Yrot;
M34=[cos(theta4) 0 sin(theta4); 0 1 0; -sin(theta4) 0 cos(theta4)];
M02=M01*M12;
M03=M02*M23;
M04=M03*M34;
%OP=simplify(OO1+M01*O1O2+M02*O2O3+M03*O3O4+M04*O4P)
l1z=0.25;
l1z=eval('l1z');
l2y=0.25;
l2y=eval('l2y');
l2z=0.75;
l2z=eval('l2z');
l3=0.25;
l3=eval('l3');
l4=0.5;
l4=eval('l4');
l5=0.5;
l5=eval('l5');
tr1=0.25;
OO1=[tr1; 0; l1z];
O1O2=[0; l2y; l2z];
O2O3=[0; l3; 0];
O3O4=[0; l4; 0];
O4P=[0; l5; 0];
OP=OO1+M01*O1O2+M02*O2O3+M03*O3O4+M04*O4P
%{
M01=[subs(Xrot, theta, theta1) [0;l2y;l2z]; 0 0 0 1];
```

```
M02=[subs(Zrot, theta, theta2) [0;l3;0]; 0 0 0 1];
M03=[subs(Xrot, theta, theta3) [0;l4;0]; 0 0 0 1];
M04=[subs(Yrot, theta, theta4) [0;l5;0]; 0 0 0 1];
%}
```

LISA 3. DENAVIIT-HARTENBERGI MEETOD

$R_{i-1} \rightarrow R_i$	θ_i	d_i	a_i	α_i
$R_0 \rightarrow R_1$	θ_1	d_1	0	α_1
$R_1 \rightarrow R_2$	θ_2	d_2	a_2	α_2
$R_2 \rightarrow R_3$	θ_3	d_3	a_3	α_3
$R_3 \rightarrow R_4$	θ_4	0	0	α_4
$R_4 \rightarrow R_5$	θ_5	1	0	α_5

$R_{i-1} \rightarrow R_i$	θ_i	d_i	a_i	α_i
$R_0 \rightarrow R_1$	90°	0,25	0	90°
$R_1 \rightarrow R_2$	0°	0,25	0,25	−90°
$R_2 \rightarrow R_3$	−45°	0,75	0,25	−90°
$R_3 \rightarrow R_4$	−60°	0	0	−90°
$R_4 \rightarrow R_5$	0°	1	0	0

$$M_0^1 = \begin{bmatrix} cos\theta_1 & -sin\theta_1 cos\alpha_1 & sin\theta_1 sin\alpha_1 & a_1 cos\theta_1 \\ sin\theta_1 & cos\theta_1 cos\alpha_1 & -cos\theta_1 sin\alpha_1 & a_1 sin\theta_1 \\ 0 & sin\alpha_1 & cos\alpha_1 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} cos90° & -sin90°\cos(90°) & sin90°\sin(90°) & 0 \cdot cos90° \\ sin90° & cos90°\cos(90°) & -cos90°\sin(90°) & 0 \cdot sin90° \\ 0 & \sin(90°) & \cos(90°) & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & d_1 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0,25 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$M_1^2 = \begin{bmatrix} cos\theta_2 & -sin\theta_2 cos\alpha_2 & sin\theta_2 sin\alpha_2 & a_2 cos\theta_2 \\ sin\theta_2 & cos\theta_2 cos\alpha_2 & -cos\theta_2 sin\alpha_2 & a_2 sin\theta_2 \\ 0 & sin\alpha_1 & cos\alpha_1 & d_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} cos0° & -sin0°\cos(-90°) & sin0°\sin(-90°) & 0{,}25cos0° \\ sin0° & cos0°\cos(-90°) & -cos0°\sin(-90°) & 0{,}25sin0° \\ 0 & \sin(-90°) & \cos(-90°) & 0{,}25 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0{,}25 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0{,}25 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$M_2^3 = \begin{bmatrix} cos\theta_3 & -sin\theta_3 cos\alpha_3 & sin\theta_3 sin\alpha_3 & a_3 cos\theta_3 \\ sin\theta_3 & cos\theta_3 cos\alpha_3 & -cos\theta_3 sin\alpha_3 & a_3 sin\theta_3 \\ 0 & sin\alpha_3 & cos\alpha_3 & d_3 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} \cos(-45°) & -\sin(-45°)\cos(-90°) & \sin(-45°)\sin(-90°) & 0{,}25 \cdot \cos(-45°) \\ \sin(-45°) & \cos(-45°)\cos(-90°) & -\cos(-45°)\sin(-90°) & 0{,}25 \cdot \sin(-45°) \\ 0 & \sin(-90°) & \cos(-90°) & 0{,}75 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} \frac{\sqrt{2}}{2} & 0 & \frac{\sqrt{2}}{2} & 0{,}25 \cdot \frac{\sqrt{2}}{2} \\ -\frac{\sqrt{2}}{2} & 0 & \frac{\sqrt{2}}{2} & -0{,}25 \cdot \frac{\sqrt{2}}{2} \\ 0 & -1 & 0 & 0.75 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$M_3^4 = \begin{bmatrix} cos\theta_4 & -sin\theta_4 cos\alpha_4 & sin\theta_4 sin\alpha_4 & a_4 cos\theta_4 \\ sin\theta_4 & cos\theta_4 cos\alpha_4 & -cos\theta_4 sin\alpha_4 & a_4 sin\theta_4 \\ 0 & sin\alpha_4 & cos\alpha_4 & d_4 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} \cos(-60°) & -\sin(-60°)\cos(-90°) & \sin(-60°)\sin(-90°) & 0 \cdot \cos(-60°) \\ \sin(-60°) & \cos(-60°)\cos(-90°) & -\cos(-60°)\sin(-90°) & 0 \cdot \sin(-60°) \\ 0 & \sin(-90°) & \cos(-90°) & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} \frac{1}{2} & 0 & \frac{\sqrt{3}}{2} & 0 \\ -\frac{\sqrt{3}}{2} & 0 & \frac{1}{2} & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$M_4^5 = \begin{bmatrix} cos\theta_5 & -sin\theta_5 cos\alpha_5 & sin\theta_5 sin\alpha_5 & a_5 cos\theta_5 \\ sin\theta_5 & cos\theta_5 cos\alpha_5 & -cos\theta_5 sin\alpha_5 & a_5 sin\theta_5 \\ 0 & sin\alpha_5 & cos\alpha_5 & d_5 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} cos0° & -sin0° cos0° & sin0° sin0° & 0 \cdot cos0° \\ sin0° & cos0° cos0° & -cos0° sin0° & 0 \cdot sin0° \\ 0 & sin0° & cos0° & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$M_0^5 = M_0^1 M_1^2 M_2^3 M_3^4 M_4^5$$

$$M_0^1 M_1^2 = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0,25 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0,25 \\ 0 & 0 & 1 & 0 \\ 0 & -1 & 0 & 0,25 \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 0 & -1 & 0 & 0,25 \\ 1 & 0 & 0 & 0,25 \\ 0 & 0 & 1 & 0,25 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$M_0^1 M_1^2 M_2^3 = \begin{bmatrix} 0 & -1 & 0 & 0,25 \\ 1 & 0 & 0 & 0,25 \\ 0 & 0 & 1 & 0,25 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{\sqrt{2}}{2} & 0 & \frac{\sqrt{2}}{2} & 0,25 \cdot \frac{\sqrt{2}}{2} \\ -\frac{\sqrt{2}}{2} & 0 & \frac{\sqrt{2}}{2} & -0,25 \cdot \frac{\sqrt{2}}{2} \\ 0 & -1 & 0 & 0.75 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} \frac{\sqrt{2}}{2} & 0 & -\frac{\sqrt{2}}{2} & 0,25 \cdot \frac{\sqrt{2}}{2} + 0,25 \\ \frac{\sqrt{2}}{2} & 0 & \frac{\sqrt{2}}{2} & 0,25 \cdot \frac{\sqrt{2}}{2} + 0,25 \\ 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$M_0^4 = M_0^1 M_1^2 M_2^3 M_3^4 = \begin{bmatrix} \frac{\sqrt{2}}{2} & 0 & -\frac{\sqrt{2}}{2} & 0{,}25 \cdot \frac{\sqrt{2}}{2} + 0{,}25 \\ \frac{\sqrt{2}}{2} & 0 & \frac{\sqrt{2}}{2} & 0{,}25 \cdot \frac{\sqrt{2}}{2} + 0{,}25 \\ 0 & -1 & 0 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \frac{1}{2} & 0 & \frac{\sqrt{3}}{2} & 0 \\ -\frac{\sqrt{3}}{2} & 0 & \frac{1}{2} & 0 \\ 0 & -1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} \frac{\sqrt{2}}{4} & \frac{\sqrt{2}}{2} & \frac{\sqrt{2}\sqrt{3}}{4} & 0{,}25 \cdot \frac{\sqrt{2}}{2} + 0{,}25 \\ \frac{\sqrt{2}}{4} & -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}\sqrt{3}}{4} & 0{,}25 \cdot \frac{\sqrt{2}}{2} + 0{,}25 \\ \frac{\sqrt{3}}{2} & 0 & -\frac{1}{2} & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$M_0^5 = M_0^1 M_1^2 M_2^3 M_3^4 M_4^5 = \begin{bmatrix} \frac{\sqrt{2}}{4} & \frac{\sqrt{2}}{2} & \frac{\sqrt{2}\sqrt{3}}{4} & 0{,}25 \cdot \frac{\sqrt{2}}{2} + 0{,}25 \\ \frac{\sqrt{2}}{4} & -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}\sqrt{3}}{4} & 0{,}25 \cdot \frac{\sqrt{2}}{2} + 0{,}25 \\ \frac{\sqrt{3}}{2} & 0 & -\frac{1}{2} & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} \frac{\sqrt{2}}{4} & \frac{\sqrt{2}}{2} & \frac{\sqrt{2}\sqrt{3}}{4} & \frac{\sqrt{2}\sqrt{3}}{4} + 0{,}25 \cdot \frac{\sqrt{2}}{2} + 0{,}25 \\ \frac{\sqrt{2}}{4} & -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}\sqrt{3}}{4} & \frac{\sqrt{2}\sqrt{3}}{4} + 0{,}25 \cdot \frac{\sqrt{2}}{2} + 0{,}25 \\ \frac{\sqrt{3}}{2} & 0 & -\frac{1}{2} & -\frac{1}{2} + 1 \\ 0 & 0 & 0 & 1 \end{bmatrix} =$$

$$= \begin{bmatrix} \frac{\sqrt{2}}{4} & \frac{\sqrt{2}}{2} & \frac{\sqrt{2}\sqrt{3}}{4} & \frac{\sqrt{2}\sqrt{3}}{4} + 0{,}25 \cdot \left(\frac{\sqrt{2}}{2} + 1\right) \\ \frac{\sqrt{2}}{4} & -\frac{\sqrt{2}}{2} & \frac{\sqrt{2}\sqrt{3}}{4} & \frac{\sqrt{2}\sqrt{3}}{4} + 0{,}25 \cdot \left(\frac{\sqrt{2}}{2} + 1\right) \\ \frac{\sqrt{3}}{2} & 0 & -\frac{1}{2} & \frac{1}{2} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$\overrightarrow{OP}_{/Ro} = \begin{bmatrix} \dfrac{\sqrt{2}\sqrt{3}}{4} + 0{,}25 \cdot \left(\dfrac{\sqrt{2}}{2} + 1\right) \\ \dfrac{\sqrt{2}\sqrt{3}}{4} + 0{,}25 \cdot \left(\dfrac{\sqrt{2}}{2} + 1\right) \\ \dfrac{1}{2} \end{bmatrix} \approx \begin{bmatrix} 1{,}039 \\ 1{,}039 \\ 0{,}5 \end{bmatrix}$$

LISA 4. DENAVIT JA HARTENBERG. KOOD

.m fail

```
function [A]=DH(a,alpha,d,theta)
A=[cos(theta) -sin(theta)*round(cos(alpha)) sin(theta)*round(sin(alpha))
a*cos(theta)
  sin(theta) cos(theta)*round(cos(alpha)) -cos(theta)*round(sin(alpha))
a*sin(theta)
  0            round(sin(alpha))              round(cos(alpha))
  d
  0                    0                              0
  1];
end
```

Programm 1

```
%TRRR-robot simulatsioon
clc
clearvars
%{
function [A]=DH(a,alpha,d,theta)
A=[cos(theta) -sin(theta)*round(cos(alpha)) sin(theta)*round(sin(alpha))
a*cos(theta)
  sin(theta) cos(theta)*round(cos(alpha)) -cos(theta)*round(sin(alpha))
a*sin(theta)
  0            round(sin(alpha))              round(cos(alpha))
  d
  0                    0                              0
  1];
end
theta d a alpha
%L(1)=Link([90 0.25 0 90], 'standart');
%L(2)=Link([0 0.25 0.25 -90], 'standart');
%L(3)=Link([-45 0.75 0.25 -90], 'standart');
%L(4)=Link([-60 0 0 -90], 'standart');
%L(5)=Link([0 1 0 0], 'standart');
L(1)=Link([pi/2, 0.25, 0, pi/2, 1]);
L(2)=Link([0, 0.25, 0.25, -pi/2, 0]);
L(3)=Link([-pi/4, 0.75, 0.25, -pi/2, 0]);
L(4)=Link([-pi/3, 0, 0, -pi/2, 0]);
L(5)=Link([0, 1, 0, 0, 0]);
%}
%syms theta1 theta2 d2 theta3 theta4 theta5 d1 d3 d4 d5 a1 a2 a3 a4
%syms d2
%DH parameeters
%Link1
a1=0; alpha1=pi/2; d1=0.25; theta1=pi/2;
%Link2
a2=0.25; alpha2=-pi/2; d2=0.25; theta2=0;
%Link3
a3=0.25; alpha3=-pi/2; d3=0.75; theta3=-pi/4;
%Link4
a4=0; alpha4=-pi/2; d4=0; theta4=-pi/3;
%Link5
a5=0; alpha5=0; d5=1; theta5=0;
H0_1=DH(a1, alpha1, d1, theta1);
H1_2=DH(a2, alpha2, d2, theta2);
H2_3=DH(a3, alpha3, d3, theta3);
```

```
H3_4=DH(a4, alpha4, d4, theta4);
H4_5=DH(a5, alpha5, d5, theta5);
H0_2=H0_1*H1_2;
H0_3=H0_2*H2_3;
H0_4=H0_3*H3_4;
H0_5=H0_4*H4_5;
%{
   x01=H0_1(1,4);
   y01=H0_1(2,4);
   z01=H0_1(3,4);

   x02=H0_2(1,4);
   y02=H0_2(2,4);
   z02=H0_2(3,4);

   x03=H0_3(1,4);
   y03=H0_3(2,4);
   z03=H0_3(3,4);

   x04=H0_4(1,4);
   y04=H0_4(2,4);
   z04=H0_4(3,4);

   x05=H0_5(1,4);
   y05=H0_5(2,4);
   z05=H0_5(3,4);

   %plot3(0,0,0,r)
   %X=[0 x01 x02 x03 x04 x05];
   %Y=[0 y01 y02 y03 y04 y05];
   %Z=[0 z01 z02 z03 z04 z05];
   X1=[0 x01];
   X2=[x01 x02];
   X3=[x02 x03];
   X4=[x03 x04];
   X5=[x04 x05];
   Y1=[0 y01];
   Y2=[y01 y02];
   Y3=[y02 y03];
   Y4=[y03 y04];
   Y5=[y04 y05];
   Z1=[0 z01];
   Z2=[z01 z02];
   Z3=[z02 z03];
   Z4=[z03 z04];
   Z5=[z04 z05];
   %Y=[0 y01 y02 y03 y04 y05];
   %Z=[0 z01 z02 z03 z04 z05];
   link1=line([0, x01], [0, y01], [0, z01]);
   link2=line([x01, x02], [y01, y02],  [z01, z02]);
   link3=line([x02, x03], [y02, y03],  [z02, z03]);
   link4=line([x03, x04], [y03, y04],  [z03, z04]);
   link5=line([x04, x05], [y04, y05],  [z04, z05]);
   %set(link1, 'LineWidth', 5, 'Color', 'red')
   %set(link2, 'LineWidth', 5, 'Color','yellow')
   %set(link3, 'LineWidth', 5, 'Color','green')
   %set(link4, 'LineWidth', 5, 'Color','magenta')
   %set(link5, 'LineWidth', 5, 'Color','black')
   %plot3(X,Y,Z)
   plot3(X1,Y1,Z1,X2,Y2,Z2,X3,Y3,Z3,X4,Y4,Z4,X5,Y5,Z5)
%}
```

```
    t=0;
   dt=0.1;
   xlim([-3,3])
   ylim([-3,3])
   zlim([-3,3])
for i=1:500
   t=t+dt;
   %d2=t;
   theta1=pi/2;
   theta2=0;
   %theta1=pi/2+sin(t);
   theta3=-pi/4+cos(t);
   theta4=-pi/3+cos(t);

   x01=0;
   y01=0;
   z01=d1;

     x02=d2*sin(theta1) + a2*cos(theta1)*cos(theta2);
     y02=a2*cos(theta2)*sin(theta1) - d2*cos(theta1);
     z02=d1 + a2*sin(theta2);
     x03=d2*sin(theta1) + a2*cos(theta1)*cos(theta2) -
d3*cos(theta1)*sin(theta2) - a3*sin(theta1)*sin(theta3) +
a3*cos(theta1)*cos(theta2)*cos(theta3);
     y03=a2*cos(theta2)*sin(theta1) - d2*cos(theta1) +
a3*cos(theta1)*sin(theta3) - d3*sin(theta1)*sin(theta2) +
a3*cos(theta2)*cos(theta3)*sin(theta1);
     z03=d1 + d3*cos(theta2) + a2*sin(theta2) +
a3*cos(theta3)*sin(theta2);
     x04=d2*sin(theta1) + a2*cos(theta1)*cos(theta2) -
d3*cos(theta1)*sin(theta2) - a3*sin(theta1)*sin(theta3) +
a3*cos(theta1)*cos(theta2)*cos(theta3);
     y04=a2*cos(theta2)*sin(theta1) - d2*cos(theta1) +
a3*cos(theta1)*sin(theta3) - d3*sin(theta1)*sin(theta2) +
a3*cos(theta2)*cos(theta3)*sin(theta1);
     z04=d1 + d3*cos(theta2) + a2*sin(theta2) +
a3*cos(theta3)*sin(theta2);
     x05=d2*sin(theta1) + sin(theta4)*(sin(theta1)*sin(theta3) -
cos(theta1)*cos(theta2)*cos(theta3)) + a2*cos(theta1)*cos(theta2) -
d3*cos(theta1)*sin(theta2) - a3*sin(theta1)*sin(theta3) +
cos(theta1)*cos(theta4)*sin(theta2) +
a3*cos(theta1)*cos(theta2)*cos(theta3);
     y05=cos(theta4)*sin(theta1)*sin(theta2) -
sin(theta4)*(cos(theta1)*sin(theta3) +
cos(theta2)*cos(theta3)*sin(theta1)) - d2*cos(theta1) +
a2*cos(theta2)*sin(theta1) + a3*cos(theta1)*sin(theta3) -
d3*sin(theta1)*sin(theta2) + a3*cos(theta2)*cos(theta3)*sin(theta1);
     z05=d1 - cos(theta2)*cos(theta4) + d3*cos(theta2) + a2*sin(theta2) -
cos(theta3)*sin(theta2)*sin(theta4) + a3*cos(theta3)*sin(theta2);
   %x01=eval('x01');
   %y01=eval('y01')

   %x02=eval('x02');
   %y02=eval('y02')
   X1=[0 x01];
   X2=[x01 x02];
   X3=[x02 x03];
   X4=[x03 x04];
   X5=[x04 x05];
   Y1=[0 y01];
   Y2=[y01 y02];
```

```
    Y3=[y02 y03];
    Y4=[y03 y04];
    Y5=[y04 y05];
    Z1=[0 z01];
    Z2=[z01 z02];
    Z3=[z02 z03];
    Z4=[z03 z04];
    Z5=[z04 z05];
    %Y=[0 y01 y02 y03 y04 y05];
    %Z=[0 z01 z02 z03 z04 z05];
    link1=line([0, x01], [0, y01], [0, z01]);
    link2=line([x01, x02], [y01, y02],  [z01, z02]);
    link3=line([x02, x03], [y02, y03],  [z02, z03]);
    link4=line([x03, x04], [y03, y04],  [z03, z04]);
    link5=line([x04, x05], [y04, y05],  [z04, z05]);
    %set(link1, 'LineWidth', 5, 'Color', 'red')
    %set(link2, 'LineWidth', 5, 'Color','yellow')
    %set(link3, 'LineWidth', 5, 'Color','green')
    %set(link4, 'LineWidth', 5, 'Color','magenta')
    %set(link5, 'LineWidth', 5, 'Color','black')
    %plot3(X,Y,Z)
    plot3(X1,Y1,Z1,X2,Y2,Z2,X3,Y3,Z3,X4,Y4,Z4,X5,Y5,Z5)

    %{
    link1=line([0, x01], [0, y01], [0, z01]);
    link2=line([x01, x02], [y01, y02],  [z01, z02]);
    link3=line([x02, x03], [y02, y03],  [z02, z03]);
    link4=line([x03, x04], [y03, y04],  [z03, z04]);
    link5=line([x04, x05], [y04, y05],  [z04, z05]);
    set(link1, 'LineWidth', 5, 'Color', 'red')
    set(link2, 'LineWidth', 5, 'Color','yellow')
    set(link3, 'LineWidth', 5, 'Color','green')
    set(link4, 'LineWidth', 5, 'Color','magenta')
    set(link5, 'LineWidth', 5, 'Color','black')

   %}
    pause(0.2)
    delete (link1)
    delete (link2)
    delete (link3)
    delete (link4)
    delete (link5)
end
```

Programm 2

```
%TRRR-robot simulatsioon
clc
clearvars
%{
function [A]=DH(a,alpha,d,theta)
A=[cos(theta) -sin(theta)*round(cos(alpha)) sin(theta)*round(sin(alpha)) a*cos(theta)
  sin(theta) cos(theta)*round(cos(alpha)) -cos(theta)*round(sin(alpha)) a*sin(theta)
  0             round(sin(alpha))                round(cos(alpha))
  d
  0                     0                                0
  1];
end
```

```
%}
syms theta1 theta2 d2 theta3 theta4 theta5 d1 d3 d4 d5 a1 a2 a3 a4
%DH parameeters
%Link1
a1=0; alpha1=pi/2; d1=d1; theta1=theta1;
%Link2
a2=a2; alpha2=-pi/2; d2=d2; theta2=theta2;
%Link3
a3=a3; alpha3=-pi/2; d3=d3; theta3=theta3;
%Link4
a4=0; alpha4=-pi/2; d4=0; theta4=theta4;
%Link5
a5=0; alpha5=0; d5=1; theta5=theta5;
H0_1=DH(a1, alpha1, d1, theta1);
H1_2=DH(a2, alpha2, d2, theta2);
H2_3=DH(a3, alpha3, d3, theta3);
H3_4=DH(a4, alpha4, d4, theta4);
H4_5=DH(a5, alpha5, d5, theta5);
H0_2=H0_1*H1_2;
H0_3=H0_2*H2_3;
H0_4=H0_3*H3_4;
H0_5=H0_4*H4_5;
x01=simplify(H0_1(1,4))
y01=simplify(H0_1(2,4))
z01=simplify(H0_1(3,4))
x02=simplify(H0_2(1,4))
y02=simplify(H0_2(2,4))
z02=simplify(H0_2(3,4))
x03=simplify(H0_3(1,4))
y03=simplify(H0_3(2,4))
z03=simplify(H0_3(3,4))
x04=simplify(H0_4(1,4))
y04=simplify(H0_4(2,4))
z04=simplify(H0_4(3,4))
x05=simplify(H0_5(1,4))
y05=simplify(H0_5(2,4))
z05=simplify(H0_5(3,4))
```

LISA 5. ROBOTICS TOOLBOX

```
clear all
close all
%L(1)=Link([90 0.25 0 90], 'standart');
%L(2)=Link([0 0.25 0.25 -90], 'standart');
%L(3)=Link([-45 0.75 0.25 -90], 'standart');
%L(4)=Link([-60 0 0 -90], 'standart');
%L(5)=Link([0 1 0 0], 'standart');
L(1)=Link([pi/2, 0.25, 0, pi/2, 1]);
L(2)=Link([0, 0.25, 0.25, -pi/2, 0]);
L(3)=Link([-pi/4, 0.75, 0.25, -pi/2, 0]);
L(4)=Link([-pi/3, 0, 0, -pi/2, 0]);
L(5)=Link([0, 1, 0, 0, 0]);
L(1).qlim=[0 0.25];
figure(1)
%L(4).offset=pi/2;
TRRR=SerialLink(L)
TRRR.name='TRRR Robot';
TRRR.plot([0 0 0 0 0]);
qf=[pi/2, -pi/6, -pi/4, pi/2, 0];
Tf=TRRR.fkine(qf);
q0=[0 0 0 0 0];
q=TRRR.ikine(Tf,q0,'mask',[1 1 1 1 1 0]);
t=0:0.15:3;
Q=jtraj(q0, qf, t);
Tr=fkine(TRRR,Q);
for i=1:1:length(t)
   T=Tr(i);
   trs=transl(T);
   xx(i)=trs(1);
   yy(i)=trs(2);
   zz(i)=trs(3);
end
plot(TRRR,Q);
hold on
plot3(xx,yy,zz,'Color',[ 1 0 0], 'LineWidth',2)
```