

Faculteit Industriële Ingenieurswetenschappen

master in de industriële wetenschappen:
elektromechanica

Masterthesis

Ontwerp en krachtgedreven aansturing van een mobiele manipulator voor het automatiseren van productietaken aan verschillende werkstations

**Bram Deboel
Stijn Loix**

Scriptie ingediend tot het behalen van de graad van master in de industriële wetenschappen: elektromechanica

PROMOTOR :

Prof. dr. ir. Eric DEMEESTER

PROMOTOR :

ing. Jan KEMPENEERS

BEGELEIDER :

ing. David DE SCHEPPER

ing. Martijn CRAMER

Gezamenlijke opleiding UHasselt en KU Leuven



Universiteit Hasselt | Campus Diepenbeek | Faculteit Industriële Ingenieurswetenschappen | Agoralaan Gebouw H - Gebouw B | BE 3590 Diepenbeek

Universiteit Hasselt | Campus Diepenbeek | Agoralaan Gebouw D | BE 3590 Diepenbeek
Universiteit Hasselt | Campus Hasselt | Martelarenlaan 42 | BE 3500 Hasselt



2024
2025

Faculteit Industriële Ingenieurswetenschappen

master in de industriële wetenschappen:
elektromechanica

Masterthesis

Ontwerp en krachtgedreven aansturing van een mobiele manipulator voor het automatiseren van productietaken aan verschillende werkstations

Bram Deboel

Stijn Loix

Scriptie ingediend tot het behalen van de graad van master in de industriële wetenschappen: elektromechanica

PROMOTOR :

Prof. dr. ir. Eric DEMEESTER

PROMOTOR :

ing. Jan KEMPENEERS

BEGELEIDER :

ing. David DE SCHEPPER

ing. Martijn CRAMER



KU LEUVEN

Woord vooraf

Een masterproef betekent het einde van de opleiding tot industrieel ingenieur. We hebben ons het voorbije academiejaar verdiept in de wereld van mobiele robots en hebben hier veel over bijgeleerd. Het project is niet altijd verlopen zoals we dachten, we hebben enkele tegenslagen gekend op het vlak van onderdelen besteld krijgen maar zijn alsnog zeer blij met het resultaat dat we behaald hebben. Er is gelachen, er waren frustraties, er is occasioneel eens kwaad geworden, kortom alles om met een goed gevoel terug te kunnen kijken op dit project was aanwezig.

We zouden hier ook graag onze oprechte dank uitspreken aan een aantal personen die een cruciale rol hebben gespeeld binnen deze masterthesis. Allereerst gaat onze dank uit naar Prof. dr. ir. Eric Demeester voor het opvolgen van het project en zijn waardevolle inzichten. Daarnaast zijn we ook Ing. David De Schepper en Ing. Martijn Cramer bijzonder dankbaar voor hun dagelijkse begeleiding en ondersteuning. Een speciale dank gaat ook uit naar ACRO voor het beschikbaar stellen van materiaal, toestellen en kennis, wat essentieel was voor het onderzoek. Ook Ing. Jan Kempeneers als klankbord bij de initiële brainstorming van het project en Prof. dr. Jeroen Lievens voor zijn ondersteuning en feedback tijdens het schrijfproces van deze scriptie. Verder zijn we ook onze ouders dankbaar voor hun financiële steun en motiverende woorden.

Als laatste wensen wij u nog veel leesplezier!

Met vriendelijke groet,

Bram Deboel

Stijn Loix

Diepenbeek, juni 2025

Inhoudsopgave

Woord vooraf	1
Lijst met tabellen	7
Lijst met figuren	11
Abstract	13
Abstract in English	15
1 Inleiding	17
1.1 Situering	17
1.2 Probleemstelling	18
1.3 Doelstelling	19
1.4 Methode	21
1.5 Vooruitblik	22
2 Bronnenstudie	23
2.1 Actieve wielmodules	24
2.1.1 Lay-out	28
2.1.2 Marktstudie	30
2.2 Frame	32
2.3 Cobot	32
2.4 Koppeling met werkstations	34
2.4.1 Stabilisatie	34
2.4.2 Dataoverdracht	34
2.4.3 Aansluitingen nutsvoorzieningen	36
2.5 Elektr(on)ische componenten	36
2.5.1 Batterij	36
2.5.2 Batterijlader	36
2.5.3 I/O-eiland	37
2.6 Veiligheidscomponenten	37
2.6.1 Noodstopknop	37
2.6.2 Veiligheidsschakelaar	37
2.6.3 Veiligheidsscanner	39
2.6.4 Veiligheidscontroller	39
2.7 Dataverwerking	40

2.7.1	Compacte controller	40
2.7.2	Industriële pc	40
2.8	Software	41
2.8.1	Besturingssysteem	41
2.8.2	Framework voor robotaansturing en dataverwerking	41
2.9	Use-cases	42
3	Conceptontwikkeling en ontwerpkeuzes	43
3.1	Wielmodules	44
3.2	Structuur en frame	45
3.2.1	Ventio-componenten	45
3.2.2	Ontwerpen	48
3.3	Cobotarm	55
3.3.1	Universal Robots UR5	55
3.3.2	Doosan H2017	56
3.4	Koppeling met werkstation	57
3.5	Resultaat	58
4	Mechanisch ontwerp en analyse	61
4.1	Situering	61
4.2	Parameterische stabiliteitsanalyse	62
4.2.1	Vaste parameters	63
4.2.2	Variabele parameters met betrekking tot het mobiele platform	63
4.2.3	Gebruikte methode	64
4.2.4	Resultaten	68
4.3	Belastingsanalyse	68
4.3.1	Krachten op actieve wielmodules	68
4.3.2	Structurele integriteit	72
5	Elektrisch ontwerp	73
5.1	Indeling	73
5.1.1	Vermogenzijde	73
5.1.2	Stuurzijde	74
5.1.3	Overige	74
5.2	Risicoanalyse	74
5.2.1	Ruimtelijke grenzen van de machine	74
5.2.2	Gebruikersgrenzen	75
5.2.3	Latente gevaren	76
5.2.4	Risico-inschatting	77
5.2.5	Risicoreductiemaatregelen en evaluatie	79
5.3	Veiligheidsconcept	79
5.3.1	Overzicht en beschrijving	79
5.3.2	Uitwerking: mechanische acties	80
5.3.3	Uitwerking: besturingstechnische acties	81
5.3.4	Uitwerking: diverse acties	81
5.4	Componenten	81
5.4.1	Veiligheid	81

5.4.2	Besturing	82
5.4.3	Communicatie	83
5.5	Elektrische schema's	83
6	Aansturing mobiel platform met twee KELO-drives	87
6.1	Werking van de bestaande ROS-code	88
6.2	Analyse	89
6.2.1	Normale operatie	89
6.2.2	Aanleggen van externe krachten	93
6.3	Aanpassingen en toevoegingen	95
6.3.1	Methode	95
6.3.2	Implementatie in code	102
7	Aansturing mobiel platform met vier KELO-drives	107
7.1	Bijkomende obstakels en functionaliteiten	107
7.2	Aanpassingen aan het algoritme	109
7.2.1	Toevoegingen aan bestaande driver code	110
7.2.2	Parameters	111
7.2.3	Structuurwijzigingen en uitbreidingen	113
7.3	Toevoegen LiDAR	121
7.3.1	Ingebruiksname	122
7.3.2	Detectie van muren	123
7.4	Testen en demonstratie van het platformgedrag	123
7.4.1	Herintegratie van twee wielmodules	124
7.4.2	Theoretische werking van vier wielmodules	132
7.5	Conclusie	133
8	Use case	135
8.1	Opstelling	135
8.2	Bevindingen	137
9	Besluit en vooruitblik	139

Literatuurlijst

A Bijlage - Parameters voor berekening kinematica en dynamica Doosan H2017

B Bijlage - Python-script voor het inlezen van de sensordata en opstellen van een map

Lijst van tabellen

1.1	Tabel van eisen	20
2.1	Conclusietabel actieve wielmodules	31
2.2	Vergelijkende tabel Doosan H2017, KUKA iiwa 14 en UR5	32
2.3	Vergelijkende tabel veiligheidscontrollers	39
3.1	KELO drive 100 specificaties	44
4.1	Vaste parameters met betrekking tot de stabiliteitsberekening	63
4.2	Variabele parameters met betrekking tot de stabiliteitsberekening	63
4.3	Variabelen en waarden voor het berekenen van de maximale hoekversnelling	65
4.4	Variabelen en waarden voor het berekenen van het minimale platformgewicht	67
4.5	Variabelen en waarden voor het berekenen van de reactiekrachten op de linkse en rechtse wielmodules	69
4.6	Variabelen en waarden voor het berekenen van de krachten verdeeld over de voorste en achterste wielmodules	71
4.7	Lasten voor de individuele wielmodules	72
5.1	Risico-inventarisatie van gevaarlijke situaties	78
5.2	Risico evaluatie na reductiemaatregelen	79
6.1	Parameters uit interne sensoren KELO drive 100	89
6.2	Eerste stap acties ziekenhuisrobot	100
6.3	Tweede stap acties ziekenhuisrobot	100
7.1	Livox MID-360 specificaties	121

Lijst van figuren

1.1	Operator aan een CNC-lasersnijder	18
1.2	Doosan H2017 collaboratieve robot	21
2.1	Schematisch overzicht bronnenstudie	23
2.2	Mecanum wiel	25
2.3	Neobotix Swerve-drive	26
2.4	KELO drive 100	27
2.5	Modellen van mobiele robotplatformen	28
2.6	Configuratie van een Mecanum-wielmodel	29
2.7	Omniwiel met rechte rollen	29
2.8	Configuraties actieve wielmodules	30
2.9	Kinematica Doosan M/H/A series (zie Bijlage A)	33
2.10	Koppelingen voor nutsvoorzieningen	36
2.11	Revolution Pi Connect, Core en Flat series	40
3.1	Mobiel platform van Sirris in de Vention-ontwerpomgeving	43
3.2	(45 x 45)mm aluminium extrusieprofiel	45
3.3	Montageonderdelen extrusieprofielen	46
3.4	Connectiemethoden extrusieprofielen	46
3.5	Zwenkwiel met uitklapbare steunpoot	47
3.6	Montageplaat voor zwenkwielen	47
3.7	Doosan H-series montageplaat	48
3.8	Eerste ontwerp voor UR5 met swerve-drives	49
3.9	Tweede ontwerp: statisch model voor UR5	50
3.10	Tweede ontwerp: swerve-drive model voor UR5	51
3.11	Tweede ontwerp: KELO model voor UR5	52
3.12	Montage twee mobiele platformen met swerve-drives	53
3.13	Voorlaatste ontwerp: groter platform met KELO-drives	53
3.14	Finale Vention-ontwerp voor Doosan H2017 met zwenkwielen	54
3.15	Bereik UR5 in relatie tot het platform	56
3.16	Bereik Doosan H2017 in relatie tot het platform	56
3.17	Penverbinding	57
3.18	Assemblage instructies pagina 12 van het mobiel platform	58
3.19	Vention platform met KELO-drives gemonteerd	59
4.1	Situatieschets stabiliteitsanalyse	62
4.2	Vrijlichaamsschema acceleratie cobotarm	65

4.3	Vrijlichaamsschema van het mobiele platform met de cobotarm in het XY-vlak . .	66
4.4	Vrijlichaamsschema van het mobiele platform met de cobotarm in het ZY-vlak . .	70
5.1	Risicograaf	77
5.2	Vention platform met aangeduide deur	80
5.3	Beckhoff EK1960 veiligheidscontroller	82
5.4	Veiligheidscomponenten	82
5.5	EtherCAT Hot Connect netwerktopologieën	83
5.6	Principeschema elektrische componenten	84
5.7	Principeschema EtherCAT-bus	85
5.8	Principeschema Ethernetnetwerk	85
6.1	Opstelling ziekenhuisrobot	87
6.2	Stand encoders hoek(rad) i.f.v. tijd(s) voor module B	90
6.3	Hoeksnelheid(rad/s) i.f.v. tijd(s) voor module B	90
6.4	Vergelijking van drie hoeksnelheid(rad/s) i.f.v. tijd(s) voor module B links	91
6.5	Kwadratuurstroom(A) i.f.v. tijd(s) voor module B	92
6.6	Hoekstand pivot(rad) en hoeksnelheid pivot(rad/s) i.f.v. tijd(s) voor module B . .	92
6.7	Hoeksnelheid(rad/s) i.f.v. tijd(s) voor module B	93
6.8	Kwadratuurstroom(A) i.f.v. tijd(s) voor module B	93
6.9	Hoeksnelheid(rad/s) i.f.v. tijd(s) voor module B	94
6.10	Kwadratuurstroom(A) i.f.v. tijd(s) voor module B	94
6.11	Finaal koppel normale operatie	96
6.12	Finaal koppel tegenwerkende kracht	96
6.13	Finaal koppel meewerkende kracht	97
6.14	Situaties ziekenhuisrobot	97
6.15	De oriëntatie van de wielmodules bij stilstand	98
6.16	Gebruiksaanwijzing ziekenhuisrobot iteratie 1	101
6.17	Flowchart algoritme iteratie 1	102
7.1	Overzicht van de interactie- en aansturingslogica van het platform	109
7.2	Oriëntaties wielmodules bij stilstand voor iteratie 2	111
7.3	Voorstelling van de zes koppels op het platform met vier wielmodules	117
7.4	Gebruiksaanwijzing ziekenhuisrobot iteratie 2	121
7.5	Livox MID-360 3D LiDAR	122
7.6	Schermopname van puntenwolken	123
7.7	Odometrisch gereconstrueerd pad voor het vierkant parcours	124
7.8	Geregistreerde koppels tijdens situaties 1–3 (voorwaartse beweging en stop)	125
7.9	Situaties 4–6: detectie en reactie bij zijwaartse en achterwaartse interactie	126
7.10	Situaties 7–9: verdere achterwaartse en linkerzijde-interactie	127
7.11	Grafieken van het voorwaarts, zijwaarts en achter koppel voor het vierkant parcours	128
7.12	Grafiek van de kwadratuurstroom voor het zijwaarts koppel van het vierkant parcours	129
7.13	Gereconstrueerd pad voor het draaiparcours	129
7.14	Situaties 10 en 11: roteren tijdens beweging naar links en rechts	130
7.15	Situaties 12 en 13: interpretatie en voortzetting van rotatie rond het centrum . . .	131
7.16	Verwachte koppels bij situaties 2, 5, 7 en 9 voor het mobiel platform met vier wielmodules	132

8.1	Opstelling use case met Doosan H2017 en bijhorende componenten	136
-----	--	-----

Abstract

KMO's in de plaatverwerkende industrie produceren vaak manueel kleine oplages van verschillende producten, wat traditionele automatisatie te complex en te duur maakt. KU Leuven-onderzoeksgroep ACRO zoekt samen met Sirris naar een oplossing hiervoor. Deze masterproef ontwikkelt daarom een mobiel platform dat operatoren assisteert bij het verplaatsen van een Doosan H2017 collaboratieve robot tussen verschillende machines om de productie te optimaliseren.

Bij aanvang werd een marktstudie naar beschikbare oplossingen uitgevoerd waarna een selectie gemaakt werd uit meerdere mechanische ontwerpen met extrusieprofielen. De stabiliteit hiervan werd geverifieerd voor een worst-case scenario om kantelen te voorkomen indien de cobot aan maximale snelheid naar boven accelereert. Dit werd gevolgd door het elektrisch ontwerp waar de geschikte actieve wielmodules, batterij en centrale controller geselecteerd werden. Hierop werd een risicoanalyse uitgevoerd om gevaren te identificeren en werden gepaste veiligheidscomponenten geselecteerd. Tot slot werd de benodigde software ontwikkeld die op basis van krachtmetingen in de wielen, de gebruiker actief helpt bij het verplaatsen van het mobiele platform.

Het resultaat is een mobiel platform dat geschikt is om een cobot op te monteren. De actieve assistentie door de gemotoriseerde wielen maakt het verplaatsen ook gemakkelijker. Deze masterproef biedt onderzoekers een goede basis om op verder te werken en kleine bedrijven een hulpmiddel om hun processen verder te automatiseren.

Abstract in English

SMEs in the sheet metal processing industry often produce small quantities of a large variety of products manually; this makes traditional automation too complex and too expensive. KU Leuven's research group ACRO is collaborating with Sirris on a solution to this problem. This master's thesis develops a mobile platform that assists operators with moving a Doosan H2017 collaborative robot between different machines to optimize production.

Initially, a market analysis was conducted to identify available solutions. Afterwards, a selection was made out of multiple mechanical designs with extrusion profiles. The stability of the designs was verified for a worst-case scenario to prevent toppling over when the cobot were to accelerate upwards at the maximum rate. This was followed by designing the electrical system where the appropriate wheel modules, battery and central controller were selected. A risk analysis was carried out to identify hazards and the appropriate safety components were selected. Lastly, software was developed to actively assist the operator with moving the mobile platform based on torque detection in the wheel modules.

The result is a mobile platform that can support a cobot. The active assist feature, in combination with the motorized wheels, enables the platform to be moved easily. This master's thesis provides researchers with a solid foundation to build upon, as well as gives small enterprises a tool to further automate their processes.

Hoofdstuk 1

Inleiding

1.1 Situering

Bedrijven met een machinepark maken producten op verzoek van hun klanten. Dit kan gaan over een paar exemplaren van dit product of honderden exemplaren. Voor de werknemers kunnen deze honderden exemplaren zeer repetitief zijn om te produceren. Voor het produceren is het machinepark van zulke bedrijven uitgerust met verschillende machines, zoals een draaibank, plooi bank, boormachines, freesmachines (figuur 1.1), lasersnijder, zaag en lastafel die al dan niet aangestuurd worden met Computer Numerical Control (CNC). Het materiaal of het tussentijdse product moet ook door de werknemers van en naar de nodige machine verplaatst worden.

Er zijn ook modernere bedrijven waar de werknemer of operator enkel de machines moet programmeren, de ruwe blok materiaal in de machine moet plaatsen, en het afgewerkte product eruit halen. Het is wel nog steeds nodig om de halffabricaten te vervoeren tussen de verschillende machines.

Het project dat hier mee aan de slag gaat is het Reconfigurable cOBotic prodUction asSistanT (ROBUST)-project. Deze heeft als doelgroep voornamelijk kleine en middelgrote ondernemingen (KMO's) die actief zijn in de plaatverwerkende industrie.

Het ROBUST-project is een samenwerking tussen Sirris, ACRO en KU Leuven. Sirris is een dynamische non-profitorganisatie, opgericht en bestuurd door de maakindustrie om deze te stimuleren. Ze helpen bedrijven hun innovatieambities waar te maken met een multidisciplinair team van 150 experts, meer dan 200 partners, ecosystemen, en een brede waaier aan industriële labo's. Op die manier realiseren ze met de bedrijven zo'n 1300 innovatieprojecten per jaar [1]. ACRO is een onderzoeksgroep verbonden aan de KU Leuven op campus Diepenbeek met vele jaren ervaring in automatisatie, computervisie en robotica. De onderzochte technologieën van ACRO zijn toepasbaar in een brede waaier van domeinen. Enkele voorbeelden hiervan zijn productie, logistiek, landbouw, zorgsector, en de bouwindustrie [2].



Figuur 1.1: Operator aan een CNC-lasersnijder

1.2 Probleemstelling

Het automatiseren van processen in een maakbedrijf met een groot aanbod aan producten, waarvan er telkens slechts enkele geproduceerd moeten worden, is momenteel een grote uitdaging in de industrie. In de huidige situatie worden deze stukken manueel gemaakt door ervaren werknemers met kennis van de verschillende, benodigde machines bij het produceren van een werkstuk. Traditionele robotarmen zijn niet geschikt in deze situaties omdat zij niet flexibel genoeg zijn en veel downtime zouden kennen. Dit komt doordat de robotarmen geherprogrammeerd moeten worden, wat ook nog extra tijd van een operator vereist. Daarnaast vereisen deze ook de kennis en expertise van een ingenieur of programmeur om ze correct te configureren. Daarom is het ook een economisch slechte beslissing om deze te implementeren.

Omwille van de schaarste op de arbeidsmarkt omtrent technisch geschoold personeel is het belangrijk dat deze mensen zo efficiënt mogelijk ingezet worden. Hierdoor is het interessant dat de productiviteit en continuïteit van een werkplaats niet direct gekoppeld zou zijn aan het aantal aanwezige werknemers met technische ervaring, zo kan er vermeden worden dat de productie stilgelegd moet worden door een gebrek aan personeel. Een ander werkpunt is het kunnen garanderen van voldoende kwaliteit van de werkstukken. Indien hetzelfde product door verschillende personen gemaakt wordt, kunnen er afwijkingen zijn; door gebruik te maken van een robot is dit te vermijden.

Een eerste grote uitdaging binnen het probleem van automatisatie in deze omgeving is dat de operator heel multifunctioneel moet zijn en moet kunnen reageren op inconsistente werkstukken. Ten tweede moet ook het transport van de werkstukken en eventuele hulpmiddelen tussen de verschillende werkstations gerealiseerd worden. Hierbij is het ontwikkelen van een goed navigatiesysteem essentieel.

1.3 Doelstelling

De hoofddoelstelling is het ontwikkelen van een mobiele manipulator die gebruikt kan worden in hoge-variatie, laag-volume productietoepassingen. Hierdoor kan de flexibiliteit van KMO's in voornamelijk de maakindustrie bevorderd worden. Een direct gevolg hiervan is dat het tekort aan geschoolde werkkrachten opgevangen kan worden, alsook dat het takenpakket van de huidige werknemers minder repetitief en manueel werk zal bevatten. Verder biedt het ook kansen om het hoofd te bieden aan fluctuerende en kleinere, meer gevarieerde bestellingen. Een belangrijk aspect hierin is dat de robot zich dient te kunnen aanpassen aan de veranderende omstandigheden alsook zich moet kunnen verplaatsen tussen de verschillende werkstations. De mobiele manipulator is opgedeeld in een mobiel platform en een collaboratieve robot (cobot) die op het platform gemonteerd wordt.

Tijdens het verplaatsen moet het mobiele platform toegang hebben tot voldoende informatie over de omgeving waarin deze zich bevindt. Deze informatie kan dan, na correcte verwerking, dienen om de aandrijving juist aan te sturen zodanig dat er geen beschadigingen aan zowel de robot zelf als aan de omgeving toegebracht worden. Verder moet het ook mogelijk zijn bepaalde routes aan te leren aan de robot door middel van een intuïtieve bediening. Deze bediening moet toelaten dat personen zonder ervaring in het trainen van robots, deze toch te laten trainen op basis van het teach-in principe. Dit principe laat toe de robot als het ware te tonen welke beweging deze moet uitvoeren om deze daarna zelfstandig te kunnen repeteren.

Er moet ook een docking procedure ontworpen worden waarbij er verschillende connecties plaatsvinden tussen de mobiele manipulator, zijn omgeving en het werkstation. Als eerste een mechanische connectie om voornamelijk het mobiele platform te stabiliseren om er zeker van te zijn dat deze zijn taken goed kan uitvoeren. Verder moet er een aansluiting zijn voor overdracht van energie en signalen tussen het werkstation en de robot. Hierbij zijn een elektrische aansluiting voor het opladen van de batterij van de robot en een persluchtaansluiting voor het bekrachtigen van eventuele eindeffectoren de voornaamste. Een andere belangrijke is dan ook nog een aansluiting die het toelaat te communiceren met het werkstation en deze te bedienen. Deze eis kunnen we als voldaan beschouwen als de verschillende overdrachten succesvol werken.

Een volledig overzicht van de eisen en voorwaarden om deze als voldaan te beschouwen, is te vinden in tabel 1.1.

Tabel 1.1: Tabel van eisen

Soort eis	Beschrijving eis	Voldaan als
Functioneel	De robotarm moet in staat zijn verschillende onderdelen op te pakken.	De eindeffector is in staat zijn 3 verschillende onderdelen te hanteleren
Functioneel	De robot moet in verplaatsbaar zijn in de werkplaats	De robot is in staat obstakels te ontwijken
Functioneel	De robot moet wendbaar zijn	De robot kan rond zijn eigen verticale middenas kan draaien
Functioneel	De robot moet aangestuurd kunnen worden op basis van het teach-in principe	Er is een mogelijkheid waarmee mensen kunnen communiceren met de robot.
Technisch	De robot moet toegang hebben tot perslucht.	Er is een persluchtaansluiting aanwezig
Technisch	De robot moet toegang hebben tot het elektriciteitsnet.	Er is een elektrische aansluiting aanwezig die gekoppeld kan worden aan het net.
Technisch	De robot moet stabiel zijn als hij aan een werkstation staat	Alle contactpunten blijven in contact met het onderoppervlak.
Technisch	Het werkblad moet toelaten makkelijk verschillende hulpstukken te fixeren	Het vastzetten van hulpstukken mag geen gereedschap vereisen.
Wens	De HMI is op basis van het active compliance principe	Er is een joystick of dergelijke aanwezig om de robot mee te bedienen

Tijdens het verloop van de masterproef zijn enkele doelstellingen aan het project toegevoegd. De verschillende iteraties van ontwerpen die in de volgende hoofdstukken besproken zullen worden, reflecteren deze wijzigingen. Een eerste bijkomende eis is dat het mobiele platform modulair uitbreidbaar moet zijn om een optimale grootte te hebben voor enkele cobotarmen. Deze eis is later weer weggehaald omdat deze te veel complexiteit aan het ontwerp toevoegde. Verder is ook de beslissing genomen te focussen op de zwaarste cobotarm beschikbaar bij ACRO, een Doosan H2017, om het ontwerp te sturen. Ook moet de mobiele manipulator eenvoudig te kunnen transporteren zijn van en naar demo locaties in een kleine bestelwagen. Hierdoor zijn er maximale buitenafmetingen opgesteld:

- een lengte van 1,25m,
- een breedte van 1,25m en
- een hoogte van 1,50m.

1.4 Methode

Een eerste stap om de doelstellingen te realiseren, is het kijken naar bestaande robots op de markt en hoe deze gelijkaardige problemen hebben aangepakt. Zo is het mogelijk om uit verschillende types robots ideeën op te doen en deze toe te passen bij het ontwerpen van de robot. Hierbij wordt er ook al rekening gehouden met eventuele praktische uitdagingen die later in het project een probleem kunnen vormen. Hierbij wordt ook al in de richting van cobots gekeken. Cobots zijn robotarmen die geschikt zijn om samen te werken met mensen, in tegenstelling tot traditionele robotarmen, die veiligheidssystemen bezitten om het bezeren van mensen te voorkomen. Een voorbeeld hiervan is de Doosan H2017, die in figuur 1.2 afgebeeld staat.

Om de cobot inzetbaar te maken op de verschillende werkstations zal er ook een mobiel platform ontwikkeld worden. Hierop kan de cobot gemonteerd worden en kan deze verplaatst worden tussen de verschillende stations. Dit platform dient te docken aan allerlei machines of werktafels in de werkplaats. Verder zal de relatieve positionering van de cobot ten opzichte van de machine/werktafel gerealiseerd worden met een camera-systeem.

Bij de volgende stap wordt er een morfologisch overzicht opgesteld waarbij er per functie/eis verschillende oplossingen gegeven worden. Wanneer er genoeg oplossingen per eis gevonden zijn, worden hieruit enkele robots samengesteld.

Om de finale beslissing te maken over het ontwerp, dient er een kosten-batenanalyse opgesteld te worden tussen de finale robots. Aan de hand van de prijs wordt er dan een keuze gemaakt tussen de robotarm die op een zelfrijdend platform gemonteerd is of de robotarm die op een statisch platform gemonteerd is dat door een andere Automated Guided Vehicle (AGV) verplaatst wordt.

Na de selectie en aankoop van de verschillende afzonderlijke onderdelen, zullen deze samengebouwd worden. Daarna zal de programmatie beginnen waarbij gebruik gemaakt zal worden van Robot Operating System (ROS).



Figuur 1.2: Doosan H2017 collaboratieve robot [3]

1.5 Vooruitblik

Deze scriptie bevat als eerste een bronnenstudie waarbij verschillende componenten, methodes en werkingsprincipes afgetoetst worden om in de latere ontwerpfase gebruikt te worden.

Vervolgens zal besproken worden hoe het mechanische ontwerp tot stand gekomen is alsook de berekeningen die verifiëren of het gemaakte ontwerp voldoende stabiel is om als basis te dienen voor een bewegende cobot. Na dit mechanisch ontwerp is er overgegaan naar het ontwerpen van het elektrisch systeem, het maken van een risicoanalyse alsook het opstellen van de risicoreductiemaatregelen en het bespreken van de acties die hiermee gepaard gaan.

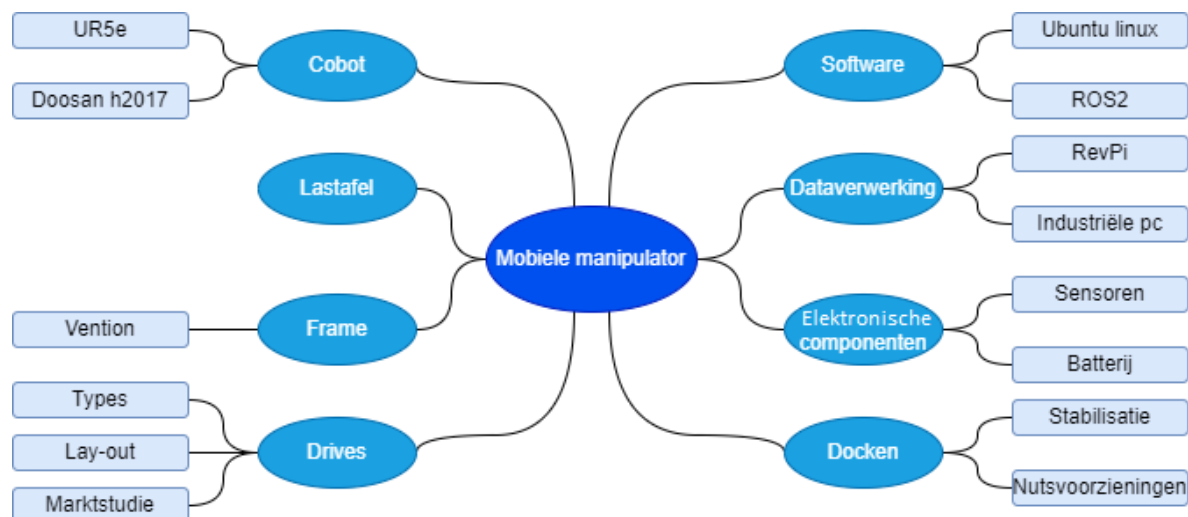
Daarna wordt de ontwikkeling van de software besproken. Deze kan opgedeeld worden in twee grote fases: de eerste iteratie is opgesteld om te dienen als testplatform voor de ontwikkeling van het controlealgoritme voor de actieve wielmodules. De tweede fase schaalt dit op naar een variabel uitbreidbare lay-out van actieve wielmodules en lost ook enkele problemen op van de eerste fase.

De scriptie sluit af met een besluit en een kritische reflectie naar wat geslaagd is, wat beter kon en welke de vervolgstappen zijn die binnen dit project gemaakt kunnen worden.

Hoofdstuk 2

Bronnenstudie

In dit deel van de thesis worden de componenten besproken die nodig zijn voor het samenstellen van het mobiel platform. Per onderdeel worden eerst alle bestaande oplossingen bekeken en ook hoe deze in verschillende usecases toegepast worden. Onderstaande figuur 2.1 geeft de inhoud van de bronnenstudie schematisch weer.



Figuur 2.1: Schematisch overzicht bronnenstudie

Eerst is er de opbouw van de mobiele manipulator, hiertoe behoren de actieve wielmodules, het frame, de lastafel en de cobot. De eigenschappen van de cobot, zoals het motorkoppel die een versnelling veroorzaakt, beïnvloeden de stabiliteit. Dit bepaalt hoe groot het frame en de lastafel dienen te zijn. In de literatuur zijn er veel verschillende cobots, hier wordt enkel onderzocht naar de aangeraden cobots voor de masterproef. De lastafel is een onderdeel dat niet zelf gefabriceerd wordt, met als gevolg dat er voor dit onderdeel geen literatuur bekeken is. De actieve wielmodules moeten in functie van de massa's van alle componenten gekozen worden zodat het totale gewicht van de mobiele manipulator gedragen kan worden.

De fijnere componenten omvatten het docken, de dataverwerking en de elektrische componenten. Deze dienen voor het besturen van de wielmodules en de cobotarm. Bij het docken dient er een mechanisme gebruikt te worden om de manipulator te stabiliseren en dienen de nodige connecties met de machines en nutsvoorzieningen gemaakt te worden.

2.1 Actieve wielmodules

Om het mobiel platform voort te kunnen bewegen, zijn er enkele belangrijke eigenschappen die in rekening gebracht worden. Om te beginnen is er de lay-out of plaatsing van de wielen. Verder zijn er de werkingsprincipes van de actieve en passieve wielen; deze hangen grotendeels samen met de lay-out. Sommige drives kunnen echter enkel functioneren met de correcte plaatsing. Daarnaast spelen andere factoren zoals de draagcapaciteit, kostprijs en gewenste snelheid een belangrijke rol in het ontwerp. De mogelijkheid om de koppels of krachten die de wielen ondervinden, uit te lezen biedt ook interessante mogelijkheden.

Eerst worden algemene mechanismes van de verschillende wielmodules vergeleken. De verkozen mechanismes, geschikt voor de toepassing van de masterproef, worden in een verder onderdeel besproken als marktstudie.

In de literatuur komen er vier verschillende types wielmodules aan bod. De twee meest voorkomende zijn de gemotoriseerde wielen en de omnidirectionele wielen. Deze bevatten typisch een enkele motor voor de aandrijving en maken gebruik van de lay-out om het mobiele platform te draaien. Er bestaan meerdere soorten omnidirectionele wielen, hiervan worden in de literatuur het vaakst mecanum-wielen gebruikt. De andere twee types zijn de swerve-drives en de KELO-drives. Bij deze zijn er typisch twee motoren aanwezig waardoor zij twee vrijheidsgraden hebben. Mobiele platformen die gebruikmaken van deze drives hebben doorgaans een hogere manoeuvreerbaarheid. De werking van elk van de vier types en hun voor- en nadelen worden verder aangehaald.

Gemotoriseerde wielen

- **Werking:** Het wiel wordt aangedreven met een motor waardoor het mobiele platform voorwaarts of achterwaarts kan bewegen. De motor kan buiten het wiel geplaatst worden met een eventuele gewenste overbrenging. Daarnaast kan de motor ook inwendig in het wiel geplaatst worden.
- **Voordelen:** Bij buiten liggende motoren is het een simpel ontwerp waardoor de kosten laag liggen. Dit systeem is zeer betrouwbaar door de simpliciteit.
- **Nadelen:** De manoeuvreerbaarheid is zeer beperkt, het is enkel geschikt voor lineaire bewegingen. De werking is ook zeer afhankelijk van het stuurmechanisme (zie 2.1.1).



Figuur 2.2: Mecanum wiel [4]

- **Werking:** Het wiel, weergegeven op figuur 2.2 bevat op de buitenkant rollers met lagers om deze vrij te laten draaien. Het ontwerp zorgt er voor dat het wiel steeds vloeiend kan rollen zonder schokken. Door de rotatiesnelheid en draairichting van deze wielen onafhankelijk aan te sturen kan het platform dankzij de rollers in elke richting uit bewegen zonder van oriëntatie te veranderen [5].
- **Voordelen:** Het ontwerp is zeer manoeuvreerbaar waardoor het ideaal is om te gebruiken in kleine ruimtes [5].
- **Nadelen:** Door de vele kleine rollers zijn er echter hoge onderhoudskosten. Doordat het wiel bestaat uit vele kleine componenten is precisie bij het produceren van belang, anders gaan er schokken tijdens het rijden optreden. De kosten om zo een wiel aan te kopen vergroten hierdoor. De ondergrond moet hierdoor evenzeer vlak en schoon zijn [6].



Figuur 2.3: Neobotix Swerve-drive [7]

- **Werking:** De swerve-drive, weergegeven in figuur 2.3, maakt gebruik van twee motoren voor het aansturen van het wiel. De eerste motor staat in voor het roteren van het wiel, de tweede voor het wiel te sturen. Elk wiel van het platform kan hierdoor onafhankelijk draaien waardoor deze in elke richting kan bewegen zonder van oriëntatie te veranderen [6].
- **Voordelen:** Zeer manoeuvreerbaar in kleine ruimtes. Doordat het geen gebruik maakt van speciale wielen is het ontwerp geschikter voor ongelijke ondergronden. In vergelijking met mecanum-wielen is het een eenvoudigere constructie. Het systeem is ook modulair waardoor de swerve-drives makkelijk vervangen kunnen worden [6].
- **Nadelen:** Deze vereisen meer ruimte op het platform door de extra benodigde motor voor de draaifunctie. Deze extra motor maakt het ontwerp ook duurder en zorgt voor toegevoegde complexiteit bij het programmeren van de drives [6].



Figuur 2.4: KELO drive 100 [8]

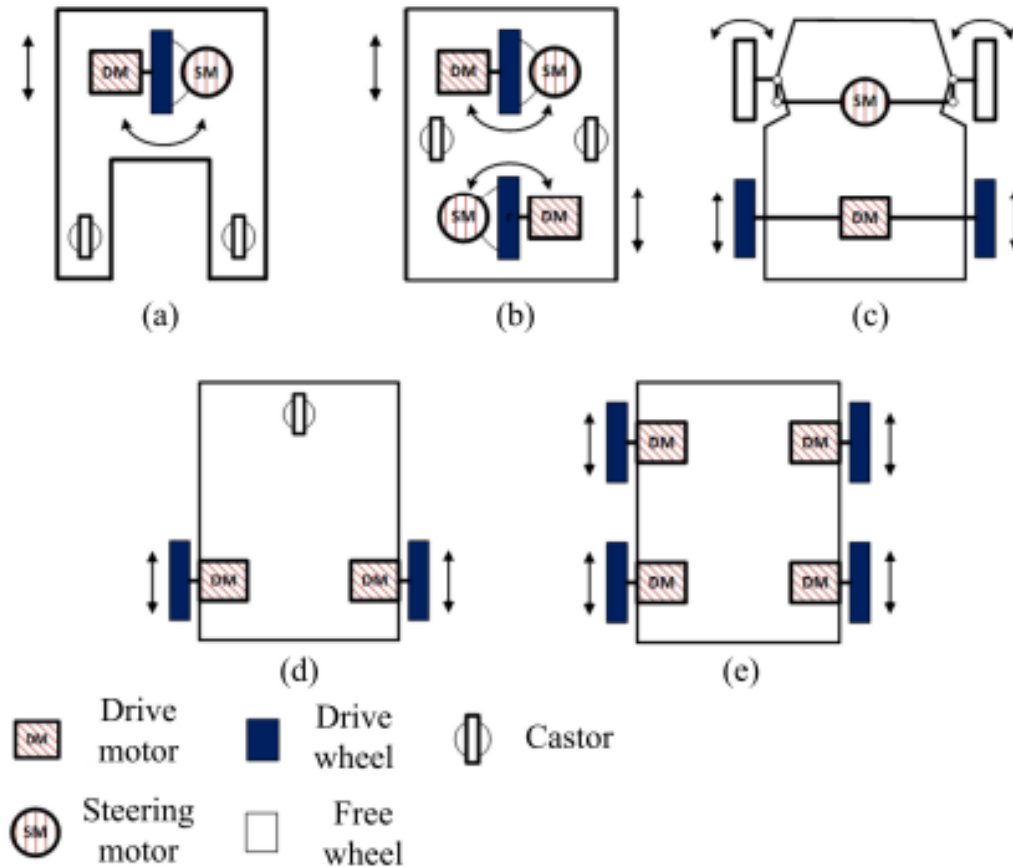
- **Werking:** Weergegeven in figuur 2.4 is een KELO-drive. Deze bestaat uit twee differentieel aangedreven wielen die vrij kunnen roteren rond hun middenas en gelijkaardig aan een swerve-drive elke gewenste oriëntatie kan aannemen. Het veranderen van oriëntatie is mogelijk door de wielen een verschillende snelheid te geven waardoor deze rond de middenas zullen draaien [8, 9].
- **Voordelen:** Net zoals de swerve-drive en de mecanum-wielen maakt de KELO-drive het mogelijk om het platform in elke richting te doen verplaatsen zonder het platform van oriëntatie te doen veranderen. Het ontwerp is ook geschikt voor zwaardere lasten [9]. Door conventionele rubberen wielen te gebruiken kan de KELO-drive ook een veel betere grip verkrijgen in tegenstelling tot mecanum-wielen [8]. Bovendien bevatten deze wielen een variëteit aan sensoren zodanig dat het koppel en de oriëntatie uitgelezen kan worden.
- **Nadelen:** Het complexe ontwerp van de KELO-drive maakt deze veel kostelijker en ook moeilijker te fabriceren [8, 9]. Beide wielen dienen ook contact te maken met de ondergrond om de KELO-unit van oriëntatie te doen veranderen. Het aansturen en programmeren van de KELO-drive is net zoals de swerve-drive complexer. Bij beweging kunnen de drives ook verstoringen van de platformpositie veroorzaken doordat er van richting verandert moet worden en de zwenkmechanismes eerst hiervoor moeten draaien [8].

Conclusie

Zowel de swerve-drive als de KELO drive 100 zijn geschikt voor industriële toepassingen, voornamelijk door hun robuuste bouw en hun prestaties op ongelijke ondergronden. Hierdoor sluiten ze goed aan op de doelstelling van deze thesis. Daarom zal de verdere marktstudie zich specifiek richten op deze twee aandrijfsystemen.

2.1.1 Lay-out

Uit de literatuur zijn er enkele configuraties voor wielmodules van mobiele platformen die vaak terugkomen, deze leiden elk tot een ander stuursysteem met andere eigenschappen en toepassingen. Een eerste paar configuraties die vooral van toepassing zijn bij gemotoriseerde wielen zijn weergegeven in figuur 2.5 en worden hieronder besproken.

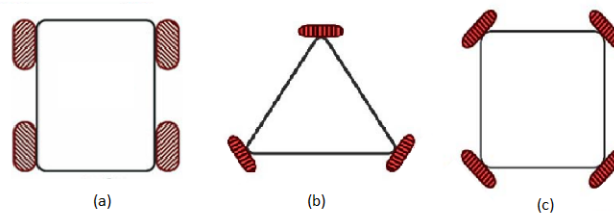


Figuur 2.5: Modellen van mobiele robotplatformen: (a) Enkele stuurbare drive; (b) twee-wiel-stuurbare drive; (c) auto gebaseerde model; (d) twee-wiel-differential drive; (e) vier-wiel-differential drive.[10, p. 109]

- **Enkele stuurbare drive:** Deze configuratie maakt gebruik van één stuurbaar aandrijfwiel om de richting en positie van de robot aan te passen. Hoewel dit systeem precisie biedt bij het volgen van eenvoudige paden, is het beperkt in zijn vermogen om van richting te veranderen wanneer het mobiele platform stilstaat.
- **Twee-wiel-stuurbare drive:** Dit systeem met een extra toegevoegd stuurbare wielen achter de passieve wielen biedt een hogere flexibiliteit in rotatie en manoeuvreerbaarheid. Het mobiele platform kan zo makkelijker ter plaatse bij stilstand van richting wisselen.
- **Auto gebaseerde model:** Dit model wordt gekenmerkt door zijn stabiliteit bij lineaire bewegingen. Doordat het niet zijwaarts kan bewegen is het echter minder geschikt voor smalle paden en omgevingen waar er snel van richting veranderd moet worden.

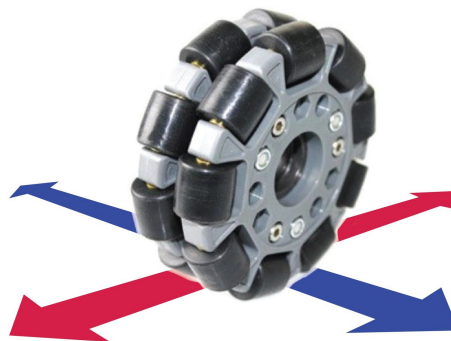
- **Twee-wiel-differential drive:** Deze configuratie maakt gebruik van twee onafhankelijk aangedreven wielen, wat resulteert in eenvoudige constructie met hoge manoeuvreerbaarheid. Afhankelijk van de montageplaatsen van deze wielen is het roteren rond de middenas van het platform al dan niet mogelijk.
- **Vier-wiel-differential drive:** Gelijkaardig aan voorgaande configuratie. De twee toegevoegde onafhankelijk aangedreven wielen geven extra manoeuvreerbaarheid, echter kunnen er nu synchronisatieproblemen optreden en is er degelijke feedback nodig.

Mecanum-wielen komen in de literatuur en in bestaande modellen onder andere configuraties voor. Onderstaande figuur 2.6 geeft drie mogelijkheden weer, verder worden deze besproken. Eigen aan mecanum-wielen is dat deze volledige holonomische bewegingsvrijheid bieden aan het platform waarop ze gemonteerd zijn [11].



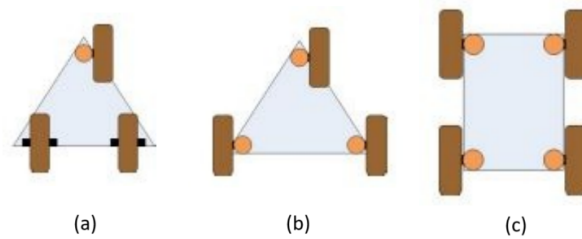
Figuur 2.6: Configuratie van een Mecanum-wielmodel: (a) parallelle configuratie; (b) drie-wiel configuratie; (c) omtrekconfiguratie [11]

- **Parallelle configuratie:** Dit is een eenvoudig ontwerp waarbij de wielen net zoals de vier wiel differentiel configuratie gestuurd worden. Deze configuratie is vooral enkel mogelijk bij mecanum-wielen doordat de rollen onder een hoek van 45° staan [11].
- **Drie-wiel configuratie:** Drie wielen zijn geplaatst in een driehoekige formatie met elk wiel onder een hoek van 120° ten opzichte van de ander.
- **Omtrekconfiguratie:** Gelijkaardig aan de parallelle configuratie met de vier wielen die differentiel aangestuurd worden. De wielen worden hier volgens de omtrek van het platform geplaatst. Dit ontwerp is eerder van toepassing bij omnidirectionele wielen waarvan de rollen loodrecht op het wiel staan zoals te zien in figuur 2.7 [11].



Figuur 2.7: Omniwiel met rechte rollen[12]

Door de gelijkende eigenschappen in werking tussen de swerve- en de KELO-drives kunnen beide gelijke lay-outs aannemen. Onderstaande figuur 2.8 geeft drie voorkomende configuraties weer.



Figuur 2.8: Configuraties actieve wielmodules: (a) één actieve en twee passieve modules; (b) drie actieve modules; (c) vier actieve modules [13, p. 44]

- **Eén actieve en twee passieve:** Door de enkele actieve wielmodule is deze configuratie zeer goedkoop. Doordat er twee passieve rollers zijn is het echter niet meer mogelijk om volledig omnidirectioneel te zijn [13, p. 44].
- **Drie actieve modules:** Deze configuratie is geschikt voor kleinere robots en laat volledige omnidirectionaliteit toe. Het is hier wel belangrijk dat de wielmodules correct geplaatst en gealigneerd worden [13, p. 44].
- **Vier actieve modules:** Dit is de meest voorkomende configuratie vanwege de hogere stabiliteit. Voor grotere en zwaardere robots is deze configuratie dus geschikter. Ook is het hier weer belangrijk om de wielmodules correct te plaatsen en te aligneren [13, p. 44].

2.1.2 Marktstudie

KELO drive 100

Op de markt is er slechts één significante fabrikant van kleinere swerve-drives die werken op basis van een differentiële aansturing, KELO-robotics. De KELO drive 100 heeft een kostprijs van 2.295 euro en een draagcapaciteit van 125 kg. Het is ook een zeer compact ontwerp doordat het gebruik maakt van wielnaafmotoren. De draagcapaciteit van 125 kg maakt het ook zeer geschikt om de KELO drive 100 te gebruiken bij industriële toepassingen en mobiele manipulators. KELO-robotics verkoopt ook speciale bouwblokken met ingebouwde KELO-drives (actief of passief), bijhorende batterijen en CPU voor een eenvoudige constructie van mobiele robots. Use cases van KELO-drives bevestigen ook dat KELO-robotics vrijwel de enige producent van dit type drives is binnen dit marktsegment [14].

Een eerste use case is van een mobiele manipulator, dit ligt zeer nauw bij het doeleinde van deze thesis. De mobiele manipulator maakt gebruik van vier KELO drives 100's. In de paper van deze usecase is er echter weinig onderzoek gedaan naar andere mogelijkheden van het drive systeem [8].

Als tweede use case is er de mobiele manipulator robot Momaro [15]. De robot maakt gebruik van vier paren van differentieel aanstuurbaar wielen, dit is hetzelfde principe als KELO-wielen. De drives die de robot gebruikt zijn echter zelf gemaakt en dus niet in die vorm beschikbaar op de markt [15].

Swerve-drives

Een eerste swerve-drive op de markt is van Swerve-drive specialties [16]. Zij verkopen verschillende kits om zelf swerve-drives te assembleren. De kits komen in verschillende vormen en hebben een ruime keuze aan motoren, die apart moeten worden aangekocht. De basisprijs van kits ligt tussen de 300 en 365 euro, de motoren hierbij kosten rond de 48 en 220 euro per motor. De draagcapaciteit van deze kits ligt echter aan de lage kant wat hun minder geschikt maakt voor zware industriële toepassingen. De producent biedt ook losse onderdelen aan waardoor het makkelijker is om deze swerve-drives te repareren [16].

Neobotix is een tweede producent van swerve-drives. Zij verkopen twee types: een OmniDrive-Module die beter kan presteren op ruwe ondergronden en een ArgoDrive die geschikter is voor compactere voertuigen. De kostprijs van beide modules is respectievelijk 7.340 en 5.000 euro en zijn beschikbaar in verschillende draagcapaciteiten tussen de 100 en 300 kg. De hogere draagcapaciteit maakt ze geschikt voor industriële toepassingen echter ligt de prijs zeer hoog [7].

De laatste producent is Yousheng uit China. Zij verkopen een breed gamma aan swerve-drives met verschillende draagcapaciteiten en motoren. Voor onze toepassing vergelijken we met het model met een draagcapaciteit van 350 kg en een kostprijs van 1.800 euro. Dit maakt het zeer geschikt voor industriële toepassingen [17].

Conclusie

Onderstaande tabel 2.1 geeft een schematisch overzicht van de verschillende mogelijke aangehaalde wielmodules die geschikt kunnen zijn voor de toepassing van de masterproef. Er wordt vergeleken op vlak van kostprijs, draagcapaciteit, en nominale spanning.

Tabel 2.1: Conclusietabel actieve wielmodules				
Specificatie	KELO drive 100	Swerve-drive kit	Neobotix	Yousheng
Kostprijs	€ 2.295	€ 400 - 550	€ 7.340	€ 1.800
Draagcapaciteit	125 kg	50 kg	150 kg	350 kg
Voltage	24/36 V	12 V	48 V	24/48 V

Voor deze masterproef zijn zowel de swerve-drive kits als de producten van Neobotix niet geschikt. De kits beschikken over een te lage draagcapaciteit, waardoor er nog extra componenten nodig zouden zijn om de draagcapaciteit op te vangen. Zowel de OmniDriveModule als de ArgoDrive hebben de goede eigenschappen maar zijn vergeleken met de ander opties, veel te duur.

De swerve-drive van Yousheng en de KELO drive 100 zijn beide goede opties. In eerste instantie lijkt de swerve-drive van Yousheng de beste optie te zijn vanwege de lagere kostprijs en hogere draagcapaciteit vergeleken met de KELO drive 100. ACRO heeft echter al meer ervaring met KELO-drives en deze vallen ook nog binnen de nodige eisen van de masterproef.

Bovendien hebben zowel Neobotix als KELO Europese verdelers waardoor de beschikbaarheid van vervangstukken en service makkelijker zal verlopen vergeleken met een Chinese producent. Dit kan een belangrijk punt zijn inzake van bedrijfscontinuïteit bij KMO's.

2.2 Frame

Voor het onderstel is er sinds het begin van de masterproef al de aanzet gegeven om deze bij het bedrijf Vention te laten maken. Zij hebben een breed gamma aan componenten, waaronder aluminium extrusieprofielen en andere vaak voorkomende onderdelen in de automatisatie industrie. Om makkelijk een ontwerp te maken beschikken zij over een online tool, de MachineBuilder, een online 3D-omgeving. In de tool kunnen alle componenten die Vention verkoopt geplaatst of gemonteerd worden. Volledige machines kunnen zo gemaakt en ook besteld worden. Het ontwerp kan ook geëxporteerd worden naar een andere omgeving zoals Inventor, waardoor visualisatie en toevoeging van overige componenten die niet bij Vention beschikbaar zijn verder mogelijk is. Vention biedt ook de extra optie aan om de machine te assembleren [18].

Aluminium extrusieprofielen zijn courant verkrijgbaar op de markt. Enkele andere producenten zijn Item, Thyssenkrupp Materials, HepcoMotion en Bosch Rexroth.

2.3 Cobot

Een collaboratieve robot (cobot) is een robot die ontworpen is om op een veilige manier repetitieve, saaie en vermoeiende taken over kunnen nemen van mensen, in omgevingen die worden gedeeld met mensen. Cobots zijn bedoeld om gebruiksvriendelijk te zijn en hebben de mogelijkheid om geprogrammeerd te worden met direct fysiek contact. Dit elimineert de nood aan een programmeur of ervaringsdeskundige doordat andere werknemers, door middel van een korte opleiding, in staat zijn hiermee te werken. Dit alles verlaagt de drempel om automatisatie te implementeren in allerhande processen. Aangezien cobots als veilig worden beschouwd, afhankelijk van de eindeffector die gemonteerd is, mogen deze gebruikt worden zonder dat extra veiligheidsmaatregelen geïmplementeerd moeten worden. Een gedetailleerde risicoanalyse hiervoor is vereist. Ook dit is een belangrijk aspect als we kijken naar de mobiele aard van het project [19].

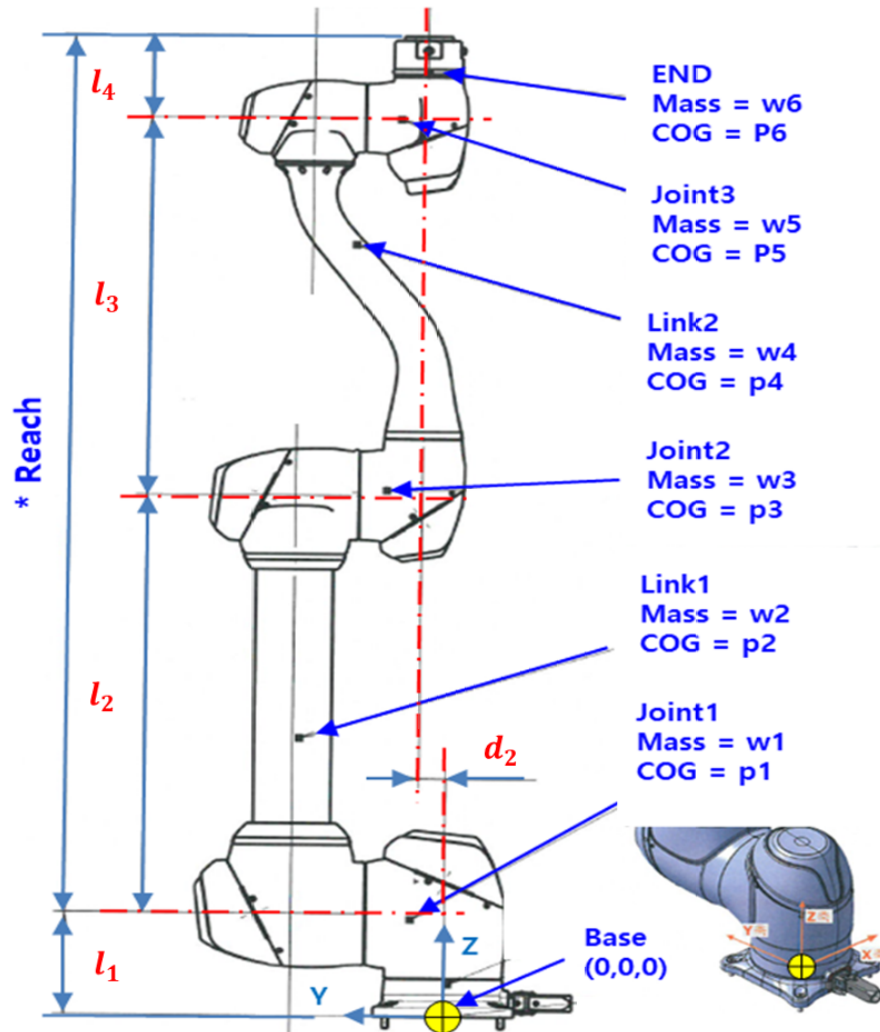
Eenmaal het mobiele platform aan een machine staat, moet de cobot hier handelingen kunnen uitvoeren. De keuze van cobot is dus een essentiële parameter bij het ontwerpen van het mobiel platform aangezien deze voldoende reikwijdte moet bezitten om de machine en onderdelen aanvoeren te kunnen bereiken. Het platform mag ook niet omkantelen als de cobot aan het bewegen is en moet dus ook stabiel genoeg zijn.

Onderzoeksgroep ACRO heeft enkele cobotarmen ter beschikking, dit omvat de UR5 cobot van Universal Robots [20], de iiwa14 van KUKA [21] en de H2017 van Doosan Robotics [3]. Tabel 2.2 geeft de meest prominente eigenschappen van de drie verschillende cobots weer. Verderop wordt er in meer detail op in gegaan.

Tabel 2.2: Vergelijkende tabel Doosan H2017, KUKA iiwa 14 en UR5 [3, 20, 21]

Eigenschap	Doosan H2017	KUKA iiwa14	UR5
Gewicht (kg)	74	29,9	18,4
Draagcapaciteit (kg)	20	14	5
Bereik (mm)	1700	820	850

De kinematica van een cobotarm komt overeen met die van een standaard robotarm. Figuur 2.9 toont de parametrisatie van de Doosan M/H/A-serie. De arm bestaat uit zeven segmenten en zes motoren, waarbij elke rotatieas wordt aangegeven met een rode punt-streep lijn.



Figuur 2.9: Kinematica Doosan M/H/A series (zie Bijlage A)

- **Base:** De basis van de cobotarm, die wordt gemonteerd op een vaste ondergrond of een mobiel platform.
- **Joint1:** Zorgt voor rotatie rond de z-as.
- **Link1:** Bevat twee rotatieassen, beide rond de y-as van de aangrenzende gewrichten.
- **Joint2:** Verantwoordelijk voor de rotatie van Link2, Joint3 en de eindeffector rond de z-as.
- **Link2:** Biedt een rotatievrijheid rond de y-as.
- **Joint3:** Zorgt voor de rotatie van de eindeffector rond de z-as.
- **END:** De eindeffector, waarop gereedschappen worden gemonteerd. Het TCP (Tool Center Point) bevindt zich op een vast punt ten opzichte van de eindeffector en bepaalt de exacte interactiepositie van het gereedschap met de omgeving.

De genoemde rotatieassen zijn gedefinieerd in een lokaal coördinatenstelsel en moeten met behulp van inverse kinematica worden omgerekend naar het basiscoördinatenstelsel. Dit proces is essentieel om de positie en oriëntatie van het TCP in de werkruimte nauwkeurig te bepalen. De Denavit-Hartenbergparameters, die de kinematische structuur wiskundig beschrijven, zijn opgenomen in de eerste tabel van Bijlage A. Daarnaast zijn de individuele massa's en zwaartepunten (COG's) van elk segment voor verschillende cobotarmen terug te vinden in de tweede en derde tabel van Bijlage A. Deze waarden worden later gebruikt bij de berekeningen voor het bepalen van de krachten die de Doosan H2017 uitoefent in een worst-case scenario.

Hoewel bovenstaande eigenschappen en parametrisatie minder relevant zijn bij het vergelijken of selecteren van een geschikte cobotarm, bieden ze wel inzicht in de opbouw en werking van een dergelijke arm. Bij de keuze van een cobotarm spelen vooral de volgende eigenschappen een doorslaggevende rol:

- **Gewicht:** Het eigengewicht van de cobotarm, gedefinieerd als de som van alle joints, links, de voet en de eindeffector, exclusief het gemonteerde gereedschap.
- **Draagcapaciteit:** Het maximale draagvermogen van de cobotarm, oftewel het gewicht dat de arm kan tillen en verplaatsen. Dit wordt vaak weergegeven in de vorm van een belastingscurve, aangezien de effectieve belasting toeneemt naarmate de last zich verder van het basispunt bevindt.
- **Bereik:** De maximale reikwijdte van de cobotarm voor het uitvoeren van taken. Dit bereik, dat begint bij de rotatie-as van Joint1 en eindigt bij het TCP, is weergegeven in figuur 2.9.

De extra toevoeging die een cobotarm speciaal maakt, zijn de koppeldetectie sensoren op elk van de motoren. Deze maken het mogelijk om eenvoudig botsingen te detecteren en op een veilige wijze in een menselijke omgeving te werken.

2.4 Koppeling met werkstations

Docken is het proces waarbij het mobiele platform zich aan het werkstation positioneert en zich voorbereidt of voorbereidt wordt om zijn taken uit te voeren. Dit proces delen we verder op in enkele stappen.

2.4.1 Stabilisatie

Wegens de substantiële massa, capaciteit en reikwijdte van de Doosan-cobot, zal het mobiele platform gestabiliseerd moeten worden tijdens zijn stationaire werkfasen. Dit moet ervoor zorgen dat ongelukken doordat het mobiele platform omvalt, vermeden worden.

2.4.2 Dataoverdracht

Om de cobot voldoende informatie te bezorgen over het werkstation waar deze momenteel aan gedocked staat, moet er data-uitwisseling gebeuren tussen het werkstation en het mobiele platform. Mogelijke data die uitgewisseld zou moeten kunnen worden is data van sensoren van bepaalde machines of het starten van bepaalde programma's bij CNC-machines.

Multi-pin connector

- **Implementatie:** Aan het werkstation worden alle sensoren, actuatoren en bedieningen aan een of meerdere kabels gekoppeld die naar een multi-pin connector vast zitten. Het mobiele platform heeft een poort die deze connector kan aanvaarden en alles kan doorkoppelen naar een I/O module op het mobiele platform om deze uit te lezen.
- **Voordelen:** Alle connecties zijn gecentraliseerd tot één connector in plaats van meerdere apart te moeten aansluiten.
- **Nadelen:** Er zijn veel punten waar een fout gemaakt kan worden bij installatie of een verbinding kan loslaten. Verder kan deze connector problemen opleveren als er vuil of vocht in de connectie zou geraken. Aangezien het aantal pinnen niet oneindig uitbreidbaar is, wordt het een complexe opgave om digitale signalen van moderne machines op deze manier door te geven. Er is veel ruimte nodig in het platform om deze signalen te ontvangen en te verwerken. Verder is het heel moeilijk om dit universeel te maken voor verschillende werkstations.

RJ45-connector

- **Implementatie:** Alle signalen die verbonden moeten worden tussen het werkstation en het mobiele platform met de cobot worden naar een centrale plaats op het station geleid en daar met een I/O module ingelezen door een microcontroller. Dit kan bijvoorbeeld een RevPi (zie sectie 2.7.1) of een EtherCAT-terminal zijn. Deze zal dan een eerste verwerking doen van de signalen en die via een bepaald protocol doorsturen over een standaard ethernetkabel naar het mobiel platform.
- **Voordelen:** Er wordt veel ruimte bespaard op het mobiele platform door de analoog-digitaal en digitaal-analoog conversies lokaal aan het werkstation te verwezenlijken.
- **Nadelen:** Componenten geassocieerd met dit type connector zijn meestal fragiel. Het gebruiken van connectoren geschikt voor industriële toepassingen zou hier een compromis kunnen zijn. Vuil in de connectoren is nog steeds een probleem.

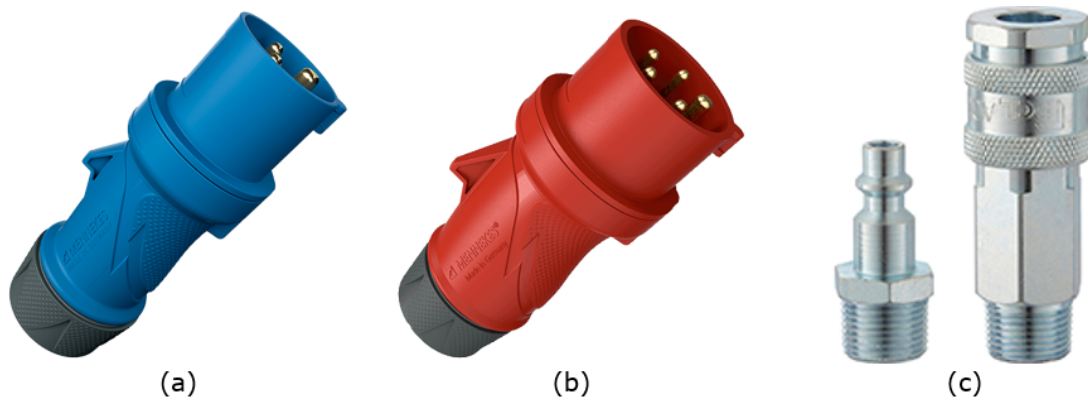
Draadloos

- **Implementatie:** Alle signalen die verbonden moeten worden tussen het werkstation en het mobiele platform met de cobot worden naar een centrale plaats op het station geleid en daar met een I/O module ingelezen door een microcontroller. Dit kan bijvoorbeeld een RevPi (zie 2.7.1) zijn. Deze zal dan een eerste verwerking doen van de signalen en die via een bepaald protocol en bluetooth of wifi doorsturen naar het mobiel platform.
- **Voordelen:** Er wordt veel ruimte bespaard op het mobiele platform door de analoog-digitaal en digitaal-analoog conversies lokaal aan het werkstation te verwezenlijken. Vuil en vocht vormen ook geen obstakel meer.
- **Nadelen:** Storingen door andere, sterke elektromagnetische golven mogelijk

2.4.3 Aansluitingen nutsvoorzieningen

Elektriciteit Er bestaan verschillende connectoren voor de overdracht van elektrische energie maar de CEE-stekker (figuur 2.10 (a) en (b)) wordt algemeen als de standaard beschouwd voor het overdragen van krachtstroom. Verder voldoen deze stekkers aan de IEC-60309 norm. Deze internationale norm legt enkele voorwaarden vast waaraan stekkers en hun accessoires aan moeten voldoen om veilig en goed voor gebruik te mogen beschouwd worden.

Perslucht Ook voor perslucht bestaan er veel connectoren. Aangezien deze allemaal op een gelijkaardig principe werken en slechts de vorm licht verschilt, kiezen we voor de ISO B12-connector (figuur 2.10 (c)).



Figuur 2.10: Koppelingen voor nutsvoorzieningen: (a) Monofase stekker [22]; (b) Driefase stekker [22]; (c) ISO B12-persluchtkoppeling [23]

2.5 Elektr(on)ische componenten

Om een operationeel platform te verkrijgen, dienen verschillende elektrische componenten toegevoegd worden.

2.5.1 Batterij

De batterij zal dienen om elektrische energie te leveren aan de mobiele manipulator tijdens het transport tussen verschillende werkstations. De capaciteit die deze moet hebben, is ingeschat door het vermogenverbruik na te gaan van de cobotarm in stilstand (om zijn positie te houden), en vier actieve wielmodules tijdens een shift van acht uur. In de praktijk zal de batterij continu opgeladen worden tijdens zijn taken aan een werkstation en kan de batterij dienen als buffer voor mogelijke piekstromen op te vangen. Met deze methode kan het verplaatsen van de mobiele manipulator altijd gebeuren met een nagenoeg volle batterij.

2.5.2 Batterijlader

Aangezien de batterij niet rechtstreeks met wisselspanning opgeladen kan worden, is een speciale gelijkrichter voor deze functie vereist. Deze heeft als bijkomende functie de batterij correct op te laden zodanig dat er geen interne cellen beschadigd raken.

2.5.3 I/O-eiland

Het I/O eiland vormt de brug tussen de componenten die een digitaal of analoog signaal geven, en de centrale controller door deze signalen om te vormen naar data dat over een veldbus verzonden kan worden. ACRO heeft reeds ervaring met een Beckhoff Automation EK1100 EtherCAT-terminal, dus deze zal ook binnen het project gebruikt worden.

Er zijn uiteraard ook andere producenten van EtherCAT-terminals. Enkele voorbeelden hiervan zijn IFM, Festo, Phoenix Contact en Wago. Een volledige lijst van compatibele producten is beschikbaar op de Product Guide van EtherCAT Technology Group [24].

2.6 Veiligheidscomponenten

In deze sectie worden alle componenten die aanwezig zijn om de veiligheid van personen in de omgeving van het mobiele platform te garanderen, besproken.

De 2006/42/EG richtlijn van de Europese Unie vermeldt: "Een machine moet zijn voorzien van één of meer noodstopinrichtingen waarmee reële of dreigende gevaarlijke situaties kunnen worden afgewend." [25, L 157/39]

2.6.1 Noodstopknop

Eén van de meest gebruikte uitvoeringen van een noodstopinrichting is de noodstopknop. Dit is in de meeste gevallen een rode drukknop met een gele achtergrond daar de richtlijn specificeert dat deze knop goed zichtbaar en herkenbaar moet zijn. Deze is te allen tijde uitgerust met ten minste één normaal gesloten contact. Wanneer de knop bediend wordt, zal deze de machine in een veilige toestand brengen.

2.6.2 Veiligheidsschakelaar

Een veiligheidsschakelaar moet worden gemonteerd op zogenoemde beweegbare afschermingen. Deze "— voorkomt dat gevaarlijke machinefuncties in werking treden totdat de afscherming gesloten is, en — een opdracht tot stopzetting geeft wanneer de afscherming niet meer gesloten is." [25, L157/43 punt 1.4.2.2]

Veiligheidsschakelaars zijn beschikbaar met een variëteit aan werkingsprincipes, elk met hun eigen voor- en nadelen die hen geschikt of ongeschikt maken voor bepaalde situaties.

Elektromechanische schakelaars

Elektromechanische schakelaars zijn te verkrijgen in verschillende uitvoeringen, zoals de standaard drukknop, trekkoordschakelaar, zwenkschakelaar of ingebouwd in het deurslot. In het geval van het deurslot kan deze ook uitgevoerd worden met een actuator die het slot sluit met een bepaalde vasthoudkracht. Deze verschillende uitvoeringen bedienen echter allemaal ten minste één verbreekcontact dat door de veiligheidssturing gedetecteerd kan worden.

RFID schakelaars

Radio Frequency Identification (RFID) is een vorm van draadloze communicatie die gebruikt maakt van elektromagnetische golven. De golflengte van deze golven liggen in het radio-gedeelte van het frequentiespectrum (3 Hz - 3000 GHz). Elk systeem dat deze technologie gebruikt, bestaat uit twee onderdelen: een lezer (zender & ontvanger) en een tag. De lezer stuurt signalen door naar de tags over elektromagnetische golven waarna de tags op eenzelfde manier antwoorden met de data die ze bezitten. De tags kunnen zogenoemd actief of passief uitgevoerd zijn. Actieve tags halen de energie die ze nodig hebben om te functioneren uit een extern systeem zoals een batterij. Passieve tags ontvangen deze energie door de magnetische golven die de lezer uitstuurt, om te vormen naar elektrische energie.

Deze technologie wordt dus ook in veiligheidsschakelaars toegepast. Enkele voordelen hieraan zijn dat de tags op meerdere manieren uitgevoerd kunnen worden. Dit biedt veel mogelijkheden op het vlak van montage. Verder kan er ook bescherming tegen manipulatie geïmplementeerd worden. Hierbij zal slechts één specifieke tag door een lezer als correct beschouwd worden. Dit voorkomt dat eender welke RFID tag kan gebruikt worden om de installatie te bedotten en alsnog in werking te stellen. Er bestaat ook de optie dit te combineren met een magnetisch slot om te verzekeren dat het onmogelijk is om de bewaakte deur te openen tijdens productie.

Deze sensoren verschillen wel van traditionele elektromechanische schakelaars op het gebied van schakelen. Deze gebruiken bewaakte halfgeleideruitgangen (OSSDs) in plaats van verbreekcontacten.

Magnetische schakelaars

Magnetische sensoren detecteren de aanwezigheid van een magnetisch veld. Dit veld kan opgewekt worden door een permanente magneet of een elektromagneet.

Dit type sensor is goed bestand tegen verontreiniging en laat ook speling toe bij de montage. Nadelen verbonden aan deze sensoren is dat ze een specifieke actuator moeten detecteren vooraleer de interne schakeling geactiveerd wordt.

Inductieve schakelaars

Inductieve sensoren baseren zich op een verandering in de veldsterkte van het magnetische veld dat deze opwekken om objecten waar te nemen. Deze verandering is het gevolg van de aanwezigheid van een metalen object in de nabije omgeving van de sensor.

Voordelen verbonden aan inductieve sensoren zijn de mogelijkheid om contactloos te functioneren. Dit heeft een significante invloed op slijtage van zowel de te detecteren objecten, als van de sensor zelf. Er is ook geen speciale actuator vereist om deze te bedienen in tegenstelling tot de RFID schakelaar, eender welk onderdeel uit een ferrometaal van de machine waarin deze is ingebouwd, volstaat. Verder is dit principe ook goed bestand tegen verontreiniging in de vorm van stof- of vloeistofdeeltjes. Dit alles draagt bij aan de uitstekende flexibiliteit die dit type sensoren over het algemeen levert.

Net zoals RFID en magnetische schakelaars gebruiken inductieve schakelaars OSSDs om veiligheidssignalen te genereren.

2.6.3 Veiligheidsscanner

De veiligheidsscanners zorgen voor een veilige zone in de nabije omgeving van het mobiele platform. De meeste types zijn gebaseerd op Light Detection And Ranging of Laser Imaging Detection And Ranging (LiDAR) technologie. Hierbij wordt er door een zender een signaal verzonden waarna dit signaal door een object weerkaatst wordt richting de ontvanger. De tijd tussen zending en ontvangst wordt gemeten waaruit men met de transmissiesnelheid van het gebruikte medium, de afstand kan afleiden.

Een interne processor in de scanner verwerkt deze meetdata en kijkt of er een object de voorgeprogrammeerde zone binnenkomt. Indien dit het geval is, kan de scanner de veiligheidscontroller verwittigen. Die verwittiging kan op verschillende methodes. Afhankelijk van de scanner kan dit over klassieke bedrading, Ethernet, EtherCAT of een ander veldbussysteem. Verder stellen de meeste scanners ook nog de rauwe meetdata ter beschikking voor andere doeleinden binnen de machine. Dit is zeker ook interessant voor het lokalisatie aspect van deze masterproef.

2.6.4 Veiligheidscontroller

De veiligheidscontroller is een secundaire processor die enkel veiligheidsgelateerde signalen verwerkt. Indien deze controller signaal krijgt van een deurcontact of van een LiDAR, zal deze de actieve wielmodules in een veilige toestand brengen. Voor het invullen van deze functie is er naar verschillende merken (PILZ, Beckhoff, Sick) en modellen gekeken, een overzicht hiervan is te vinden in tabel 2.3. Een essentiële vereiste voor de controller is dat deze ook op het EtherCAT-netwerk kan communiceren. Enerzijds om de KELO drive 100 modules te bedienen tijdens verplaatsing, anderzijds om veiligheidssignalen van het werkstation ook in te lezen en het mobiele platform in een veilige staat te brengen wanneer deze aan een werkstation gekoppeld is en een noodsituatie zich voordoet.

Tabel 2.3: Vergelijkende tabel veiligheidscontrollers [26, 27, 28, 29]

Eigenschap	PILZ PNOZmulti2	Beckhoff EK1960	Sick Flexi Soft	Sick Flexi Compact
Aantal ingangen	8	20	8	20
Aantal uitgangen	4	24 standaard 4 optioneel	4 veilige 2 test	4 veilige 8 test
Modulair uitbreidbaar	Ja	Nee*	Ja	Ja
EtherCAT compatibel	Mits module	Standaard	Mits module	Mits module

*Het is mogelijk extra EtherCAT-terminals bij te plaatsen en de veilige I/O's met de EK1960 te controleren.

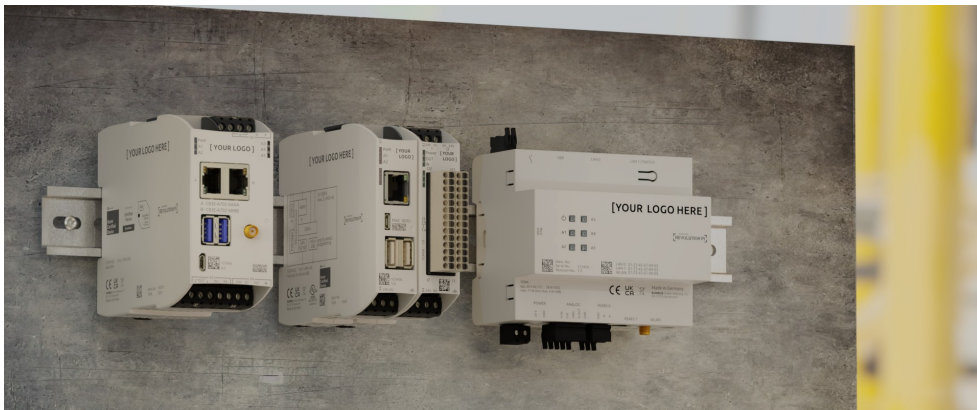
2.7 Dataverwerking

Voor het verwerken van alle data die de sensoren genereren en het aansturen van actuatoren, is er een controller nodig. Voor dit project is een platform dat compatibel is met Linux (Ubuntu) vereist.

2.7.1 Compacte controller

De Revolution Pi of RevPi afgekort is een controller gebaseerd op het Raspberry Pi platform en ontwikkeld met industriële toepassingen in het achterhoofd. Het biedt een robuuste uitvoering, DIN-rail montagemogelijkheden, 24 volt voedingsspanning alsook een modulair systeem om het platform uit te breiden met input/output (I/O) modules en extra ethernet modules. Het grote voordeel van deze controller is dat er een constante groei is aan ingenieurs die reeds in contact gekomen zijn met het platform. Hierdoor is er een grote open-source community beschikbaar waar men beroep op kan doen voor hulp. [30] Het nadeel aan de RevPi is dan weer dat deze gelimiteerd is op het vlak van rekenkracht. Dit kan mogelijk een probleem vormen indien kalibratie op basis van camerabeelden geïntegreerd moet worden.

De RevPi is echter wel geschikt om de communicatie zoals beschreven onder sectie 2.4.2 aan de kant van het werkstation te faciliteren, voor draadloos in het bijzonder. In deze toepassing wordt deze ook gebruikt in het mobiele platform dat door Sirris ontwikkeld wordt.



Figuur 2.11: Revolution Pi Connect, Core en Flat series [31]

Naast de RevPi zijn er nog verschillende andere compacte controllers die voor deze toepassing overwogen kunnen worden. Hierbij kan er gedacht worden aan de Factor-serie van OnLogic alsook de LOGO! van Siemens. Deze laatste is echter niet compatibel met het EtherCAT-protocol.

2.7.2 Industriële pc

Om meer rekenkracht en opslag ter beschikking te hebben, kan er gekozen worden om te werken met een industriële computer. Deze systemen kunnen typisch volledig naar wens geconfigureerd en aangepast worden en zijn dus zeer geschikt voor heel specifieke toepassingen. Het configureren van een pc zit niet in deze masterproef. Bij industriële pc's is er ook de mogelijkheid om de zwakste component, de ventilator, achterwege te laten door het gebruik van een voldoende grote koelvinnen. Verder bieden zo ook veel mogelijkheden op het gebied van sensoren, internetconnectie, gebruik van cloud services en het beheren van data [32].

2.8 Software

In dit deel worden enkele belangrijke software besproken die essentieel zijn voor de werking van het mobiele platform.

2.8.1 Besturingssysteem

De computer die gebruikt zal worden om het mobiele platform aan te sturen, zal Ubuntu Desktop gebruiken als operating system. Ubuntu is een open-source, Debian-based Linux besturingssysteem (OS). Het is ontwikkeld in 2004 door Canonical Ltd. met als doel een sociaal en economisch platform te creëren, beschikbaar voor iedereen, aan dezelfde voorwaarden. Het was ook het eerste OS dat de belofte deed om consistent elke zes maanden sinds oktober 2004 een nieuwe versie uit te brengen. Sinds 2006 is er beslist dat elke vierde versie, voor lange tijd ondersteund zal worden voor grootschalige projecten. Ubuntu desktop is ondertussen ook 's werelds meest gebruikte platform. [33]

De keuze voor Ubuntu is gedreven door het gebruik van ROS (zie 2.8.2) om het mobiele platform in zijn totaliteit te besturen.

2.8.2 Framework voor robotaansturing en dataverwerking

Het Robotic Operating System (ROS) is open-source framework ontwikkeld door Open Robotics dat onderzoekers en ontwikkelaars helpt code te bouwen en te hergebruiken tussen verschillende applicaties binnen robotica. Het is onderhand geëvolueerd naar ROS2, een complete herontwikkeling van het framework om de tekortkomingen geassocieerd met de eerste versie weg te werken. Het platform wordt gebruikt door enkelen van de grote namen binnen robotica zoals ABB en Doosan. Doordat ROS2 zo courant beschikbaar is doorheen de industrie, groeien de mogelijkheden die het biedt ook constant.

Enkele voordelen die het biedt, zijn:

- Kennis en structuur van de code zijn universeel voor verschillende platformen. Denk hierbij aan drones, mobiele platformen of robotarmen.
- Het platform ondersteunt meerdere programmeertalen. Nodes kunnen zowel in C++ als Python geschreven worden en nog steeds met elkaar communiceren.
- Het platform maakt het creëren van digital twins van reële toepassingen makkelijk. Dit laat het toe om snel prototypes te ontwikkelen en te testen [34] [35].

2.9 Use-cases

In het kader van het project waar deze masterproef in kadert, is er ook een bedrijfsbezoek uitgevoerd bij Opsinox in Nazareth. Zij produceren allerhande onderdelen voor rookgasafvoer- en HVAC-installaties [36]. Eigen aan hun productieproces is dat er telkens kleine oplages geproduceerd worden uit een klein assortiment aan producten. Met andere woorden, de doelgroep van het project.

Tijdens dit bedrijfsbezoek zijn er verschillende mogelijke toepassingen voor dit project naar voor gekomen. Een eerste is bijvoorbeeld eenvoudige plaatjes plooien. Hierbij zou de cobot een plaatje uit een voorraadmagazijn kunnen nemen, hier enkele plooibewerkingen op uitvoeren en het geplooidde plaatje in een ander magazijn kunnen plaatsen, klaar voor volgende bewerkingen. Hier zou dan ook een koppeling met de plooibank voorzien moeten worden zodanig dat de mobiele manipulator deze kan bedienen.

Eenzelfde procedure zou plaatsvinden bij een persmachine waar een bepaalde vorm over een mal moet gelegd worden om deze uit te rekken. De specifieke persmachine van Opsinox is echter een ouder type met primitieve bedienorganen. Dit is een uitstekend voorbeeld dat benadrukt hoe belangrijk het is dat de mobiele manipulator in staat moet zijn samen te werken met een breed gamma aan andere productiemachines.

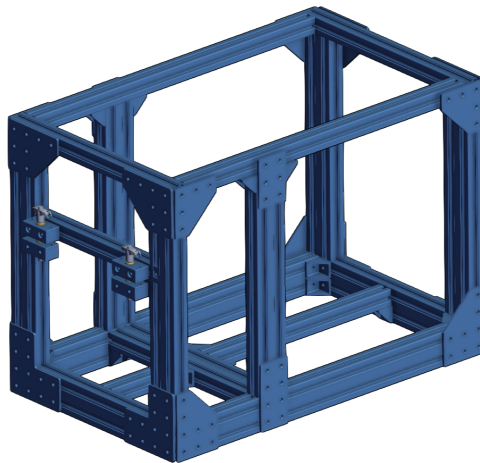
Puntlassen is een derde toepassing die door Opsinox naar voor gebracht werd. Dit is echter al een complexere procedure waarbij momenteel het onderdeel met twee handen vastgenomen wordt en doorheen het proces enkele keren gewisseld wordt van grip.

Als laatste toepassing werden eenvoudige lastaken voorgesteld. Ook dit is een complex proces aangezien het onderdeel voor en na het lassen verplaatst moet worden alsook gefixeerd met hulpstukken.

Hoofdstuk 3

Conceptontwikkeling en ontwerpkeuzes

Om te beginnen met het opstellen van een concept is het handig om te kijken naar het bestaande ontwerp van het mobiele platform van Sirris. Deze heeft gelijkaardige doeleinden als deze masterproef. Hier is ook het doel om een cobotarm op te monteren. Deze is echter bedoeld voor lichtere cobots, en wordt ingezet om taken te laten uitvoeren aan verschillende werkstations. Enkele van deze taken houden in het bedienen van een CNC-machine of een boormachine. Hun ontwerp, zoals te zien op onderstaande figuur 3.1, plaatst de profielen in de vorm van een balk met extra steunprofielen aan de zijkanten. Deze steunprofielen worden voornamelijk gebruikt om het mobiele platform in twee te verdelen, waarbij dan het onderscheid gemaakt wordt tussen een netspanning en lage veiligheidsspanning. Bovenop deze structuur van profielen is een lastafelblad op maat gemaakt. Dit blad steekt ook aan de randen uit om zo de LiDAR-sensoren die zij gebruiken te beschermen bij een eventuele botsing.



Figuur 3.1: Mobiel platform van Sirris in de Vention-ontwerpomgeving

Om het platform rond te bewegen maakt het platform van Sirris gebruik van zwenkwielen die geplaatst zijn op elk van de hoeken zodat het platform handmatig verplaatst kan worden. Deze zwenkwielen bevatten ook ingebouwde poten die uitgeduwd kunnen worden om zo, wanneer het platform op de gewenste locatie is, dit stabiel op deze locatie kan blijven.

3.1 Wielmodules

Om het platform te bewegen is er gekozen om dit zowel actief als passief te laten gebeuren. Het is dus mogelijk om de zwenkwielen uit te wisselen met actieve wielmodules. Dit is de keuze van het bedrijf afhankelijk van waar het kan toegepast worden. Uiteraard is het gebruik van zwenkwielen goedkoper dan de actieve wielmodules. Voor de wielmodules is er gekozen voor KELO-wielen. Dit zijn twee differentieel aangestuurde wielen die ook nog gemonteerd staan op een zwenkverbinding. Dit type wielen maakt het mogelijk om volledig omnidirectioneel te sturen, zo lang de kinematica van het platform dit toe laat. De KELO's zijn ook gekozen omdat de onderzoeksgroep ACRO hier al ervaring mee heeft. Enkele belangrijke eigenschappen van de KELO-drive die nodig zijn bij verdere ontwerp- en componentenkeuze zijn weergegeven in onderstaande tabel 3.1.

Tabel 3.1: KELO drive 100 specificaties [14]

Specificatie	Waarde
Draagcapaciteit	125 kg
Omgeschrevecirkeldiameter	190 mm
Hoogte	160 mm
Spanning	24 V
Stroom	20 A
Communicatie	EtherCAT

Verder bevatten de KELO-drives ook nog enkele sensoren. Ten eerst zijn er de optische encoders, hiervan zijn er drie in het totaal waarvan er twee zich bevinden in de wielen en één in de zwenkverbinding. De optische encoders laten toe om de roterende beweging te meten, deze doen dit aan de hand van lichtpulsen. Dit gebeurt aan de hand van een schijf die gemonteerd staat op de roterende as, op deze schijf bevinden zich op reguliere afstanden gleuven. Centraal staat een lichtbron en aan de tegengestelde kant van de gleuven een optische sensor. Doordat de schijf gaat roteren neemt de optische sensor lichtpulsen waar, dit bepaalt dan de rotatiesnelheid van de as [37, pp. 20-22].

Als tweede zijn er IMU's die zes vrijheidsgraden kunnen beschrijven. Deze omvatten de lineaire versnelling en hoekversnelling in de x-, y- en z-as. Het detecteren van de lineaire versnelling gebeurt aan de hand van accelerometers. Deze versnelling wordt bepaald aan de hand van het verschil tussen elektrische capaciteit. Op een centraal gewicht dat kan veren staat een uitsteeksel. Dit bevindt zich in rust centraal tussen twee elektroden, samen vormt dit een differentiële condensator. Bij beweging gaat het uitsteeksel zich meer naar een elektrode begeven, hierdoor verandert de capaciteit die de condensator geeft. Deze verandering wordt dan omgerekend naar de lineaire versnelling [38].

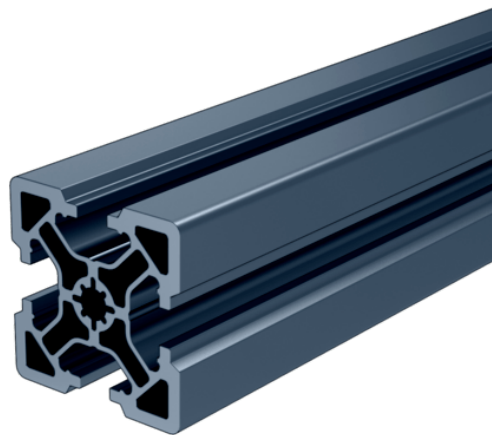
Voor het detecteren van de hoekversnelling worden gyroscopen gebruikt. Deze maken gebruik van het Corioliseffect. In deze sensor wordt een massa in trilling gebracht, zodra er rotatie optreedt zorgt het Corioliseffect voor een tweede trilling. Deze wordt dan ook door een differentiële condensator en verandering aan capaciteit opgemeten [38].

Deze sensoren laten ook toe om het krachtkoppel te detecteren, zo kan wanneer er tegen het platform gedrukt wordt een kracht worden waargenomen. Dit laat toe om de wielen op het geleverde koppel in te werken en zo, of tegen te werken en het platform op zijn plaats te houden, of om mee te gaan en in de gewenste richting voort te bewegen.

3.2 Structuur en frame

3.2.1 Vention-componenten

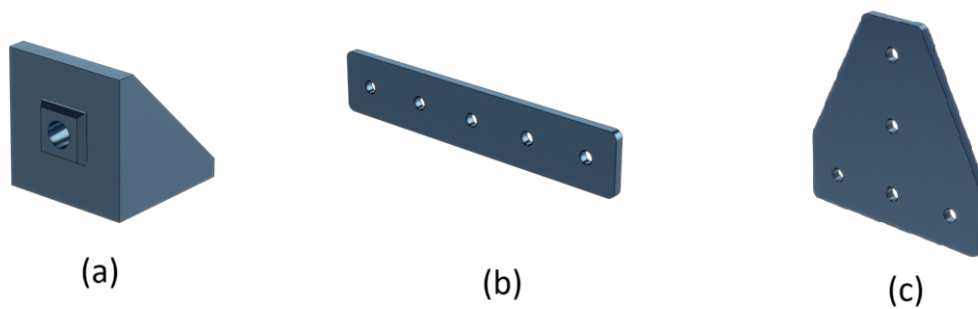
Voor de structuur en het frame is er gekozen voor een gelijkaardige aanpak als Sirris. Zij hebben hun mobiel platform ontworpen bij het bedrijf Vention. Dit is voor de masterproef sinds het begin ook als doelstelling gezet om het platform bij Vention te ontwerpen. Deze doelstelling leidt er toe dat enkele eigenschappen vastliggen. Als eerst zorgt dit er voor dat de materiaalkeuze vast ligt. De profielen die Vention aanbiedt zijn gemaakt uit aluminium 6005-T5. Deze aluminiumlegering is makkelijk te extruderen en levert ook een mooi afgewerkt oppervlak. Ten tweede ligt zo ook de profielkeuze vast. Zij produceren aluminium extrusieprofielen met weergegeven doorsnede in figuur 3.2. De profielen hebben standaard afmetingen van 45 mm hoog en 45 mm breed, de lengte van de profielen is bestelbaar voor lengtes in stappen van 45 mm. Bij het ontwerpen van het frame moet er dus rekening mee gehouden worden dat de lengte, breedte en hoogte veelvouden zijn van 45 mm. Naast de basisafmetingen zijn er nog bredere profielen met afmetingen van (45 x 90)mm en (90 x 90)mm voor zwaardere constructies [18].



Figuur 3.2: (45 x 45)mm aluminium extrusieprofiel [18]

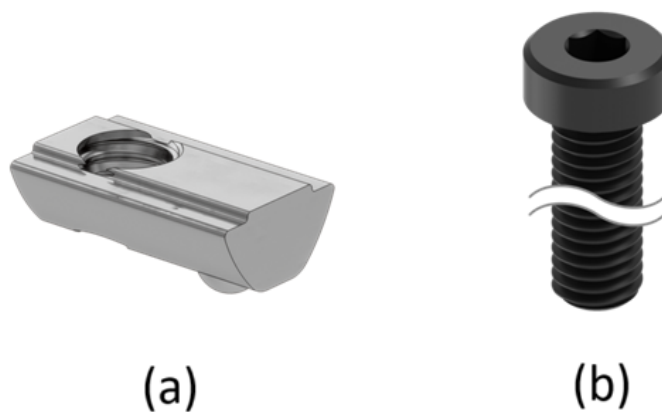
Om de profielen aan elkaar vast te monteren biedt Vention ook een breed gamma aan producten. Een eerste standaardmogelijkheid zijn hoekstukken weergegeven op figuur 3.3 (a). Ook deze zijn gemaakt uit aluminium. Om de montage te vergemakkelijken zijn er zelfpositionerende locators voorzien rondom de schroefgaten, die het hoekstuk centraal boven de gleuf uitlijnen en vermijden dat het hoekstuk kan roteren tijdens de montage. Dit hoekstuk heeft de fabrikant ook in verschillende vormen specifiek voor andere vaak voorkomende situaties [18].

Een tweede mogelijkheid om de profielen aan elkaar te monteren zijn assemblageplaten. Deze biedt Vention aan in voornamelijk twee soorten vormen. Eerst is er de rechte vorm weergegeven in figuur 3.3 (b), deze is gemaakt uit de aluminium legering 6061-T6. Daarnaast zijn er assemblageplaten in de vorm van een T weergegeven in figuur 3.3 (c), gemaakt uit dezelfde aluminiumlegering [18].



Figuur 3.3: Montageonderdelen extrusieprofielen: (a) hoekstuk, (b) rechthoekige assemblageplaat, (c) T-vormige assemblageplaat [18]

Om de bovenstaande componenten samen te voegen, wordt er gebruik gemaakt van een M8 Drop-in veerbelaste T-moer (a) en een M8 x 18 mm schroef (b), beide zijn weergegeven in onderstaande figuur 3.4. De T-moer kan dankzij de veer langs boven in de gleuf van een profiel geduwd worden en vastgeklikt. Het is mogelijk om deze in de lengte van de gleuf vrij te verplaatsen [18].



Figuur 3.4: Connectiemethoden extrusieprofielen (a) M8 Drop-in veerbelaste T-moer, (b) M8 x 18 mm schroef [18]

Omdat Vention een uitgebreid gamma aan overige componenten heeft zoals wielen, pneumatica, accessoires, veiligheid,... zijn verder enkel nog relevante componenten beschreven die terugkomen bij de verschillende ontwerpen.

Wielen

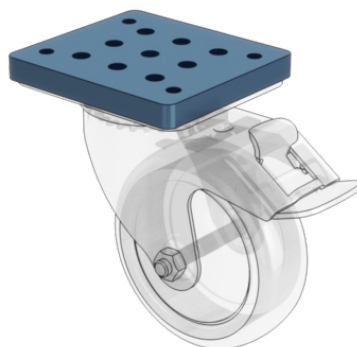
De wielsoorten die Vention aanbiedt zijn zwenkwielen met of zonder uitklapbare steunpoot. Het gebruikte zwenkwiel in de verschillende ontwerpen is met uitklapbare steunpoot te zien op figuur 3.5. Hier is een pedaal voorzien wat de steunpoot naar onder duwt waardoor het wiel tot 2 mm boven de grond gebracht wordt. Dit wiel heeft ook een laadcapaciteit van 200 kg [18].



Figuur 3.5: Zwenkwiel met uitklapbare steunpoot [18]

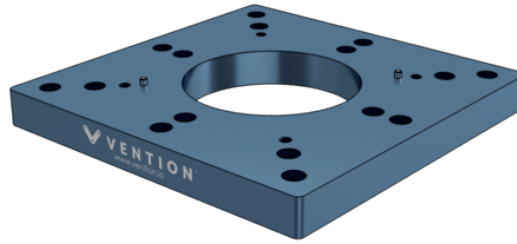
Montageplaten

De montage platen zijn door Vention speciaal ontworpen om componenten zoals cobot-armen of wielen gemakkelijk met het extrusieprofiel principe te connecteren. Een eerste montageplaat is deze voor bovenstaande zwenkwielen van figuur 3.5, deze plaat is te zien in figuur 3.6. Voor elk zwenkwiel is er echter geen unieke montageplaat, maar eerder een plaat die ontworpen is voor verschillende zwenkwielen [18].



Figuur 3.6: Montageplaat voor zwenkwielen [18]

Daarnaast zijn er de montage platen voor de cobot-armen. Deze zijn beschikbaar voor zowel elke grote cobotfabrikant waaronder Doosan, KUKA, Universal Robots. Onderstaande figuur 3.7 geeft de montageplaat weer voor de H-serie aan cobot armen van Doosan. Ook voor de cobot-armen heeft Vention de montageplaten multifunctioneel gemaakt en kunnen meerdere verschillende armen gemonteerd worden op dezelfde plaat [18].



Figuur 3.7: Doosan H-series montageplaat [18]

Afscherming

Om het binnenste gedeelte van het mobiele platform af te kunnen schermen heeft Vention ook de mogelijkheid om panelen te configureren. In de digitale ontwerpomgeving kunnen de buitenste profielen geselecteerd worden en de tool dan gebruiken om zo een paneel aan te maken dat dit gebied omvat. Deze tool houdt ook rekening met eventuele assemblageplaten waardoor er overlap voorkomen wordt. De gebruiker kan zelf ook nog kiezen waar er gaten geplaatst moeten worden indien er zich bijvoorbeeld een extra profiel in het midden bevindt waar het paneel ook op bevestigd moet worden. De panelen zijn in verschillende materialen beschikbaar waaronder HDPE, PMMA, staal en aluminium. Vanwege kostprijs en gewicht is er bij het finale ontwerp gekozen voor panelen gemaakt uit HDPE en PMMA [18].

3.2.2 Ontwerpen

Om bovenstaande componenten samen te voegen tot een definitief ontwerp, biedt Vention een intuïtieve en gebruiksvriendelijke machinebuilder aan. Dit online platform stelt gebruikers in staat om de verschillende componenten eenvoudig op te roepen en ze visueel samen te voegen tot een functioneel geheel. Elk onderdeel wordt nauwkeurig geplaatst volgens de ontwerpregels van het systeem, wat zorgt voor een gestroomlijnde en foutloze samenstelling.

Een belangrijk aspect van de machinebuilder is de manier waarop aluminium extrusieprofielen van (45 x 45) mm verwerkt worden. De bevestigingspunten zijn standaard geplaatst op een raster van 45 mm, en de profielen zelf zijn enkel beschikbaar in lengtes die een veelvoud zijn van deze maat. Dit zorgt voor een modulaire en efficiënte werkwijze, maar in uitzonderlijke gevallen kan er, op vraag van de klant en met input van de ingenieurs, worden afgeweken van deze standaard. Hierdoor blijft er flexibiliteit bestaan voor specifieke ontwerpvereisten.

Naast de standaardontwerpmogelijkheden biedt de machinebuilder ook uitgebreide CAD-integraties. Ontwerpen kunnen eenvoudig geëxporteerd worden als CAD-model, waardoor ze verder bewerkt kunnen worden in externe software. Daarnaast is het mogelijk om bestaande CAD-modellen te importeren voor componenten die niet standaard in het Vention-ecosysteem beschikbaar zijn. Tijdens het importeren kunnen gebruikers zelf de bevestigingspunten kiezen. Dit zorgt ervoor dat externe componenten naadloos geïntegreerd kunnen worden in het ontwerp, met dezelfde precisie en eenvoud als de standaard Vention-onderdelen.

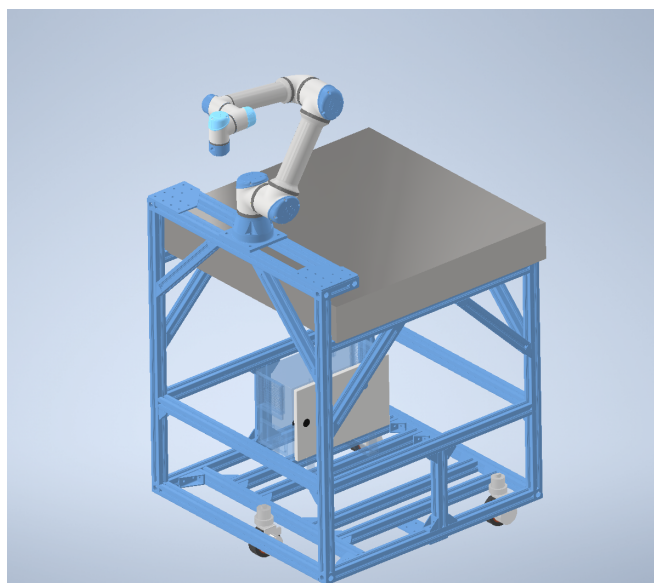
Wanneer het ontwerp is afgerond, genereert de machinebuilder automatisch een gedetailleerde componentenlijst. Hierin staan alle gebruikte onderdelen vermeld, inclusief hun specificaties en hoeveelheden. Bovendien worden real-time de kostprijs, de assemblagetijd en het totale gewicht berekend. Dit biedt direct inzicht in de haalbaarheid en praktische uitvoering van het ontwerp. Ook wordt de verwachte levertermijn weergegeven, zodat gebruikers hun projectplanning hierop kunnen afstemmen.

Indien er behoefte is aan extra ondersteuning, biedt Vention ook ingenieursondersteuning. De controle van het ontwerp kan volledig automatisch verlopen via de software, waarbij het systeem mogelijke problemen detecteert en suggesties doet. Voor complexere vragen of maatwerkoplossingen kan er ook persoonlijk contact worden opgenomen met een specialist via een vergadering. Dit hybride ondersteuningsmodel zorgt ervoor dat zowel standaard als op maat gemaakte ontwerpen efficiënt en nauwkeurig worden gevalideerd.

De verschillende ontwerpen worden onderstaand kort aangehaald. Er is telkens beschreven waarom er niet voor dat ontwerp gekozen is en wat aangepast moet worden. Zo is er een evolutie zichtbaar dat de grote basis aan mogelijkheden leidt tot het uiteindelijke finale ontwerp.

Eerste ontwerp

Onderstaande figuur 3.8 geeft het eerste ontwerp weer. De doelstellingen waren op dit punt nog niet concreet, ook was het ontwerp van Sirris nog niet gekend dus was er nog geen referentie om het ontwerp op te baseren.



Figuur 3.8: Eerste ontwerp voor UR5 met swerve-drives

De hoofdgedachte bij dit ontwerp was om dit zo goedkoop mogelijk te maken. Daarom is er voor de drives gekozen voor een driehoekige lay-out met drie swerve-drives. Uiteraard maakt dit het platform zeer onstabiel en gevoelig voor links en rechts te kantelen, wat een van de argumenten is waarom hier niet voor gekozen is en toekomstige ontwerpen vier drives bevatten.

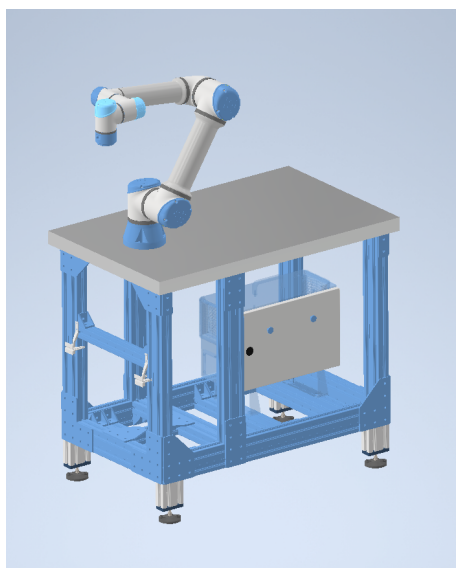
Voor het lastafelblad is er één gebruikt die commercieel verkrijgbaar is zodat dit niet zelf gemaakt moet worden. Nadelen zijn dat deze lastafel rond de 100 kg weegt en dus vrij zwaar is. Ook is er weinig flexibiliteit op de markt en zijn er weinig formaten beschikbaar. Dit maakt dat naar verder ontwerpen toe er niet meer gekozen wordt voor een commercieel verkrijgbare lastafel maar voor andere opties die bij verdere delen aangehaald zijn.

Op het ontwerp is er ook nog een aparte montageplek voorzien voor de cobotarm. Dit wordt ook aangepast naar verdere ontwerpen omdat er hierdoor geen flexibiliteit is in waar de cobotarm kan gemonteerd worden. Zo is het gewenst om deze op de lastafel te kunnen monteren en niet op een vastgezet punt.

Tussentijdse ontwerpen

De volgende serie ontwerpen is tot stand gekomen na kennismaking met het bestaande mobiele platform van Sirris. Dit oorspronkelijke ontwerp diende als referentiepunt, waardoor de nieuwe ontwerpen, weergegeven in figuur 3.9, 3.10 en 3.11, veel overeenkomsten vertonen met het originele concept. Ondanks deze gedeelde basis vertonen de drie ontwerpen significante verschillen, voornamelijk op het gebied van de aandrijving, omdat er in deze ontwerpfase nog verschillende keuzemogelijkheden open lagen.

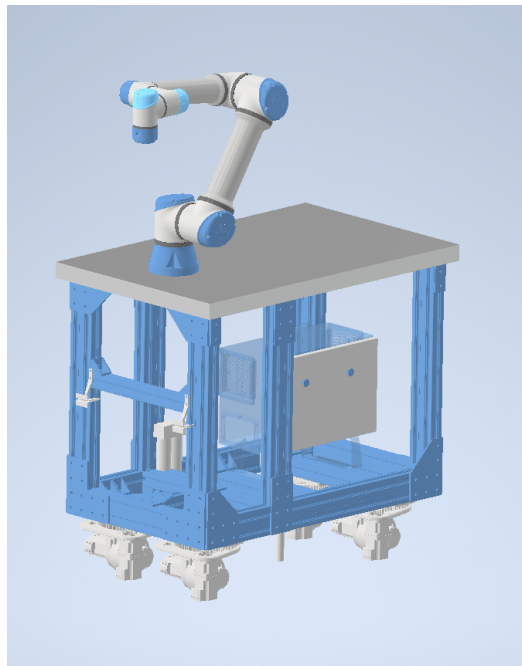
Het eerste ontwerp, het statische model (figuur 3.9), wijkt het meest af van een traditionele mobiele oplossing. In plaats van een eigen aandrijving beschikt dit platform over vaste poten. De verplaatsing ervan gebeurt niet door ingebouwde motoren of wielen, maar door een externe robot die het platform optilt, transporteert en op de gewenste locatie weer neerzet. Dit ontwerp is specifiek opgesteld om een kostenvergelijking te maken met de andere twee varianten, die respectievelijk gebruik maken van KELO- en swerve-drives.



Figuur 3.9: Tweede ontwerp: statisch model voor UR5

De vergelijking richt zich vooral op de schaalvoordelen van een gecentraliseerde verplaatsingsmethode. De kernvraag hierbij is vanaf welk aantal mobiele platformen het economisch voordeliger wordt om de individuele drives weg te laten en in plaats daarvan één afzonderlijke robot in te zetten voor de verplaatsing. Dit alternatief zou bijvoorbeeld kunnen dienen om drie of meer platformen te transporteren, wat in bepaalde scenario's een efficiëntere en kosteneffectievere oplossing zou kunnen zijn.

In onderstaande figuur 3.10 zijn de statische poten uit het vorige ontwerp vervangen door swerve-drives. Dit ontwerp maakt gebruik van een ander type swerve-drive dan degene die bij het eerste ontwerp in overweging werd genomen. De gekozen aandrijvingen bieden een aanzienlijk hogere laadcapaciteit van 350 kg en hebben als bijkomend voordeel dat ze eenvoudiger te monteren zijn onder het platform.

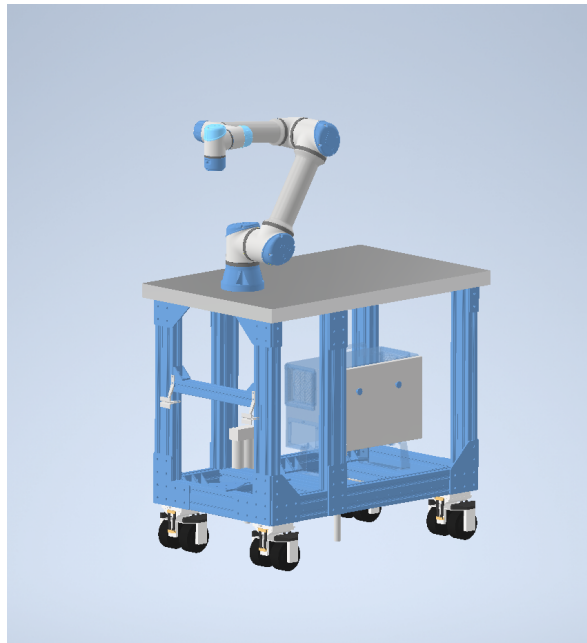


Figuur 3.10: Tweede ontwerp: swerve-drive model voor UR5

Ondanks deze verbeteringen is er een bijkomende nood aan stabilisatie, aangezien het gebruik van swerve-drives de stabiliteit van het platform tijdens operatie van de cobot-arm beïnvloedt. Om dit probleem op te lossen, zijn er lineaire actuatoren geïntegreerd in het ontwerp. Deze actuatoren worden ingezet tijdens stilstand om het platform op zijn plaats te houden, maar ze zijn niet noodzakelijk ontworpen om gewicht of krachten van de wielen op te vangen.

Daarnaast is er aan de voorzijde van het platform een zelfsluitende koppeling voorzien. Dit mechanische vergrendelingssysteem werkt door twee delen in elkaar te schuiven, waardoor een stevige verbinding ontstaat. Het ontkoppelen moet echter manueel gebeuren.

Een variant op bovenstaand ontwerp wordt weergegeven in figuur 3.11, waarin de swerve-drives vervangen zijn door KELO 100 drives. Functioneel blijven de stabilisatieoplossing met lineaire actuatoren en de zelfsluitende koppeling identiek aan de swerve drive-versie. Het belangrijkste onderscheid tussen beide ontwerpen zit in de aandrijving: de KELO 100 drives bieden een alternatieve mobiliteitsoplossing met mogelijk andere manoeuvreereigenschappen.



Figuur 3.11: Tweede ontwerp: KELO model voor UR5

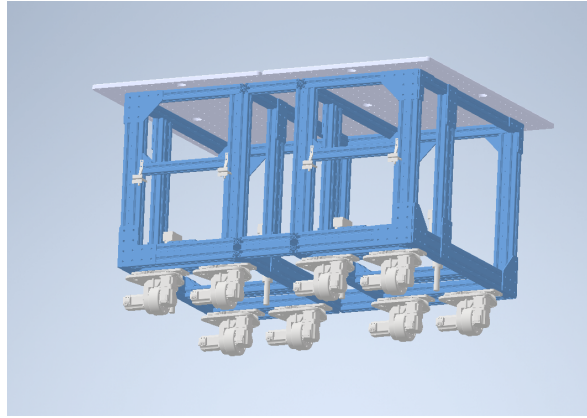
Hoewel de bovenstaande ontwerpen verschillende benaderingen verkennen, zijn ze uiteindelijk niet geselecteerd voor de integratie van de Doosan H2017-cobotarm. De voornaamste reden hiervoor is dat de platformen te klein zijn om deze robotarm correct te kunnen monteren en te laten functioneren. Dit vraagt om een alternatieve oplossing waarbij het ontwerp wordt opgeschaald of aangepast.

Daarnaast moet er een herziening plaatsvinden van de stabilisatieoplossing. Hoewel de huidige ontwerpen gebruikmaken van lineaire actuatoren tijdens stilstand, is het mogelijk dat er een robuustere of efficiëntere methode nodig is om een stabiele werkomgeving voor de cobotarm te garanderen. Dit betekent dat verdere iteraties en aanpassingen noodzakelijk zijn om tot een optimaal ontwerp te komen.

Om de integratie van de Doosan H2017-cobotarm op een mobiel platform mogelijk te maken, is het ontwerp in figuur 3.12 opgesteld. In dit concept worden twee mobiele platformen, oorspronkelijk ontworpen voor UR5-cobotarmen, aan elkaar gekoppeld om voldoende ruimte en stabiliteit te creëren voor de grotere H2017-arm.

Om een stevige verbinding tussen de twee platformen te garanderen, is ervoor gekozen om extra profielen te gebruiken. Andere methodes van koppelen bleken instabiel of onvoldoende robuust om als één geïntegreerd mobiel platform te functioneren. In het huidige ontwerp zijn acht swervedrives voorzien, maar het is mogelijk om de vier centrale aandrijvingen te verwijderen zonder de mobiliteit significant te beïnvloeden.

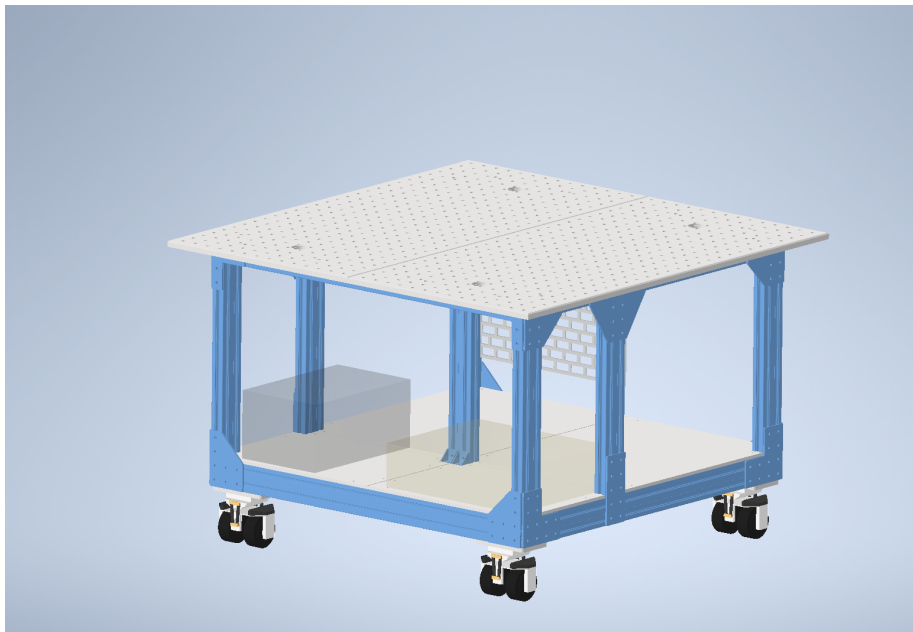
Hoewel het bovenstaande ontwerp technisch uitvoerbaar is, bleek de manier waarop de twee platformen aan elkaar gemonteerd worden te arbeidsintensief. Daarom is besloten om dit principe te laten vallen en in plaats daarvan verdere ontwerpen te ontwikkelen met een enkel platform dat van meet af aan geschikt is voor de Doosan H2017.



Figuur 3.12: Montage twee mobiele platformen met swerve-drives

Onderstaande figuur 3.13 toont het voorlaatste ontwerp, waarin de eerdere concepten zijn samengevoegd tot één enkel platform. In deze fase werd de definitieve keuze gemaakt om de KELO drive 100 te gebruiken als aandrijving. De voornaamste reden voor deze keuze was de bestaande ervaring met deze aandrijvingen binnen de betrokken onderwijsinstelling, wat de verdere ontwikkeling en integratie van het ontwerp kon vergemakkelijken. Het verminderen van onbekende variabelen in de test- en implementatiefase draagt bij aan een efficiënter ontwerpproces en lagere risico's op onverwachte complicaties.

Hoewel het ontwerp in deze versie grotendeels compleet is qua afmetingen en configuratie, ontbreekt er op dit punt nog een oplossing voor de stabilisatie en de connectie met machines. Dit betekent dat verdere verfijning nodig is om het platform geschikt te maken voor de uiteindelijke toepassing.

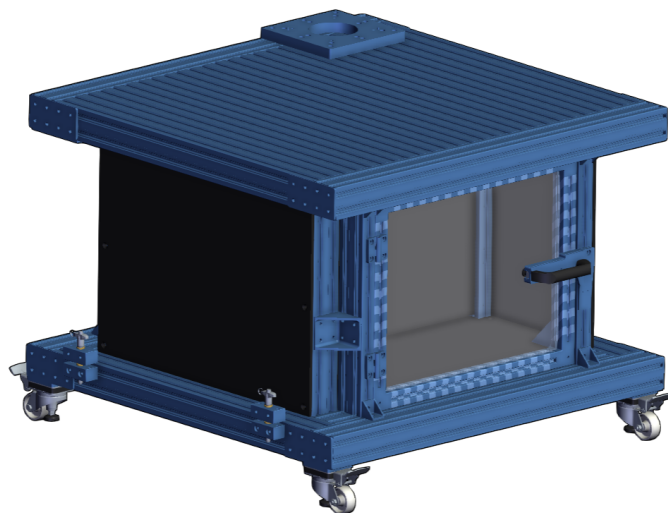


Figuur 3.13: Voorlaatste ontwerp: groter platform met KELO-drives

Dit ontwerp is verder verfijnd naar het finale ontwerp met de aandachtspunten naar het lastafblad toe en algemene versteviging van het frame. Een extra vereiste is hier ook bijgekomen dat het mobiele platform in een bestelwagen moet passen, deze met een breedte van 1,25 meter. Er moet ook nog rekening gehouden worden met de nodige grootte van de basis van de wielen en ook zodat deze wielen niet overbelast worden. De berekeningen met betrekking tot de wielbasis en belasting zijn verderop hoofdstuk 4 gedetailleerd uitgewerkt. Deze analyse was cruciaal om ervoor te zorgen dat het uiteindelijke ontwerp niet alleen functioneel was, maar ook duurzaam en veilig in gebruik.

Finale ontwerp

Het finale ontwerp is tot stand gekomen door een opeenvolging van iteraties en verfijningen op basis van eerdere ontwerpen en aanvullende eisen. Dit eindontwerp is weergegeven in figuur 3.14. Naast de interne optimalisaties werd er ook samengewerkt met het team van specialisten bij Vention. Op basis van de eerdere ontwerpversie, de uitgevoerde berekeningen en de gestelde eisen, hebben zij aanvullende verstevigingen en optimalisaties doorgevoerd om het platform te verbeteren.



Figuur 3.14: Finale Vention-ontwerp voor Doosan H2017 met zwenkwielen

Een van de belangrijkste aanpassingen in het finale ontwerp is de tijdelijke vervanging van de KELO-drives door passieve zwenkwielen met steunen, vergelijkbaar met die op het mobiele platform van Sirris. Deze tijdelijke oplossing werd gekozen om de assemblage voort te zetten terwijl er gewacht werd op de levering van de KELO-drives. Zodra deze beschikbaar zijn, worden de zwenkwielen vervangen, zodat het platform volledig operationeel wordt met de beoogde aandrijving. Daarnaast is er in het gehele ontwerp gekozen voor dikkere en stevigere profielen, wat bijdraagt aan een hogere structurele integriteit en duurzaamheid van het platform.

Om het platform stevig op zijn plaats te houden, zijn er specifieke componenten voorzien met gaten waarin pennen gestoken kunnen worden. Dit zorgt voor een mechanische fixatie die voorkomt dat het platform ongewenst verschuift tijdens stilstand of gebruik. De precieze indeling van de componenten die in de compartimenten komen, wordt verder besproken in hoofdstuk 5.

Voor extra veiligheid en bescherming zijn de compartimenten aan de binnenzijde afgeschermd met panelen en voorzien van afsluitbare deuren. Dit voorkomt niet alleen stof- en vuilophoping, maar biedt ook een afsluitbare ruimte voor waardevolle of gevoelige apparatuur.

In het finale ontwerp is ervoor gekozen om geen traditioneel, stalen lastafelblad te gebruiken, maar een oppervlakte te realiseren met Vention-extrusieprofielen. Deze keuze biedt dezelfde flexibiliteit bij montage van gereedschappen en cobotarmen, maar heeft als bijkomend voordeel een eenvoudiger bestelproces door de nood aan een extra leverancier te elimineren. Bovendien is een lastafelblad aanzienlijk zwaarder en zou het speciaal op maat gemaakt moeten worden, wat extra kosten en complexiteit met zich meebrengt. Door gebruik te maken van profielen met standaardafmetingen, die reeds elders in sectie 3.2.1 besproken zijn, blijft het ontwerp licht, modulair en eenvoudig aanpasbaar zonder concessies te doen aan stabiliteit of functionaliteit.

De eerder geïntroduceerde overhang is behouden gebleven om de sensoren te beschermen. Daarnaast is aan de onderzijde van het platform een extra overhang toegevoegd. Dit vergroot de basis waarop de aandrijvingen worden gemonteerd, wat bijdraagt aan een betere stabiliteit. Tegelijkertijd is ervoor gezorgd dat de totale breedte van het platform binnen de vereiste 1,25 meter blijft, zodanig dat de mogelijkheid om het in een kleine bestelwagen vervoerd kan worden, behouden blijft.

3.3 Cobotarm

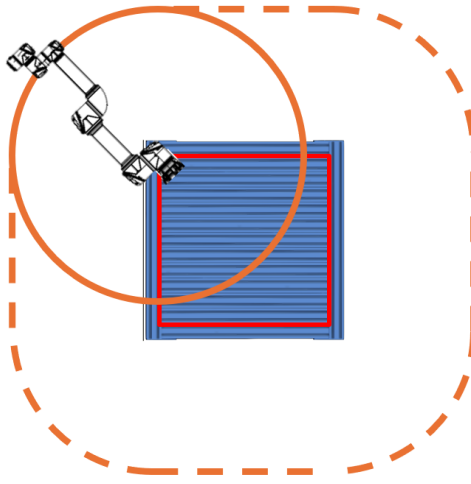
Het finale ontwerp van het mechanische platform biedt de mogelijkheid om verschillende cobots te monteren. In deze sectie worden de relevante eigenschappen besproken voor de montage van twee specifieke cobotarmen. De nadruk ligt op de reikwijdte van de arm en de afmetingen van de montagevoet, die bepalend zijn voor het mogelijke montagegebied. Deze eigenschappen worden ook gebruikt in de verdere berekeningen van het hoofdstuk.

3.3.1 Universal Robots UR5

Voor het bepalen van het mogelijke montagegebied van de UR5-arm zijn twee eigenschappen van belang:

- De reikwijdte van de arm bedraagt 850 mm, volgens de datasheet van Universal Robots [20].
- De arm wordt gemonteerd via een montageplaat van (180×180) mm, geleverd door Vention [18].

Aangezien het rotatiecentrum van de arm zich in het midden van de montagevoet bevindt, ligt dit punt op 90 mm van elke rand. Hierdoor ontstaat een theoretisch mogelijk montagegebied van (990×990) mm waarin het middelpunt van de montagevoet geplaatst kan worden. In figuur 3.15 is dit gebied weergegeven als een rood vierkant. De maximale reikwijdte is aangeduid in het oranje.



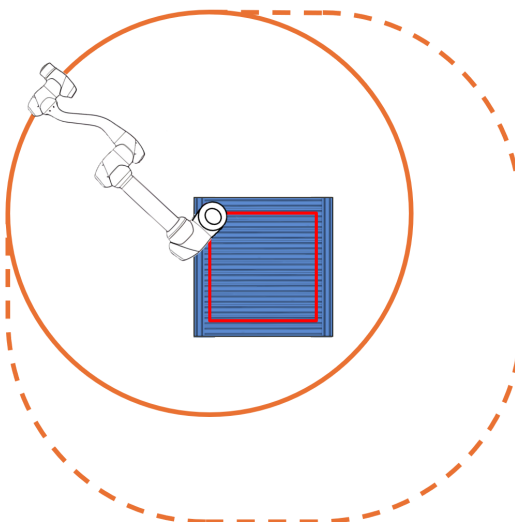
Figuur 3.15: Bereik UR5 in relatie tot het platform

Het platform bestaat uit een raster van aluminium extrusieprofielen van 45 mm, wat betekent dat de mogelijke montageposities in stappen van 45 mm verschuiven. Hierdoor is het effectieve aantal bruikbare montageposities beperkt tot een aantal discrete vierkanten binnen het totale gebied.

3.3.2 Doosan H2017

De Doosan H2017 heeft een grotere reikwijdte van 1700 mm [3]. De montagevoet van deze arm meet (268×268) mm, waardoor het middelpunt van de voet 134 mm van elke rand verwijderd is [18]. Dit resulteert in een maximaal montagegebied van (902×902) mm waarin het middelpunt van de arm geplaatst kan worden.

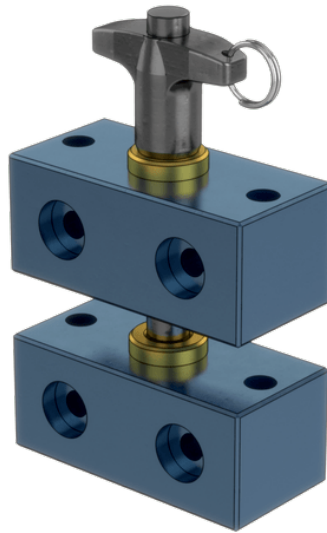
Figuur 3.16 toont dit gebied als een rood vierkant, met opnieuw de maximale reikwijdte aangeduid in oranje. Ook hier geldt dat de rasterstructuur van 45 mm tussen de extrusieprofielen de mogelijke montageposities bepaalt.



Figuur 3.16: Bereik Doosan H2017 in relatie tot het platform

3.4 Koppeling met workstation

Voor het nauwkeurig positioneren van het mobiel platform ten opzichte van een workstation is gekozen voor een penverbinding, weergegeven op figuur 3.17. Na uitgebreid onderzoek en het afwegen van verschillende opties (zie hoofdstuk 2), bleek deze methode het meest geschikt op vlak van eenvoud, reproduceerbaarheid en kostenefficiëntie.



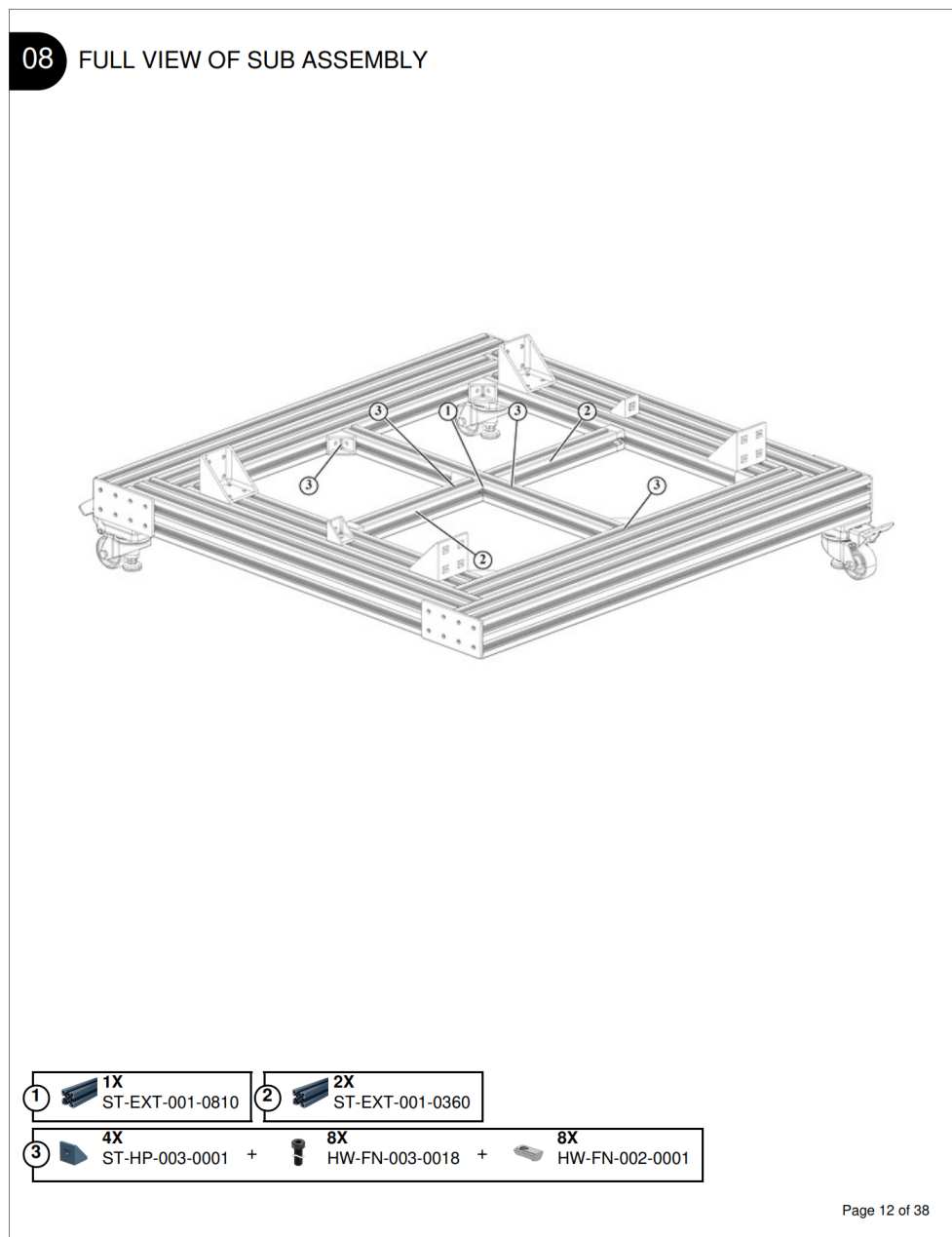
Figuur 3.17: Penverbinding [18]

De verbinding bestaat uit een tweedelig geleidingssysteem waarbij het onderste deel bevestigd wordt aan het mobiel platform, en het bovenste aan de machine. Wanneer de respectievelijke gaten van beide onderdelen uitgelijnd zijn, wordt een borgpen manueel ingebracht. Deze vergrendeling garandeert een vaste en herhaalbare positionering van het platform, cruciaal voor interactie met vaste installatiecomponenten zoals werkoppervlakken of robotarmen [18].

Daarnaast wordt de stroomvoorziening manueel aangesloten door de operator via een netsnoer. Er werd bewust gekozen om dit proces niet te automatiseren, omwille van de hoge bijkomende kost en omdat er bij elke verplaatsing van het platform toch reeds een operator aanwezig moet zijn. Deze persoon is dan ook verantwoordelijk voor het tot stand brengen van zowel de mechanische vergrendeling als de elektrische verbinding. Het is ook gevaarlijk indien het docken met 230V mis zou gaan, dus zorgt de operator ook voor extra veiligheid.

3.5 Resultaat

Na de finale goedkeuring van het Vention-ontwerp, is dit besteld. Er is voor gekozen om dit op ACRO te assembleren, zo worden er extra kosten bespaard. Om de assemblage vlot en makkelijk te laten verlopen maakt Vention standaard instructies. Een voorbeeld pagina van deze instructies is weergegeven in figuur 3.18.



Figuur 3.18: Assemblage instructies pagina 12 van het mobiel platform

Figuur 3.19 geeft het geassembleerde platform met de KELO drive 100 wielmodules gemonteerd op de hoekpunten weer.



Figuur 3.19: Vention platform met KELO-drives gemonteerd

Om de wielmodules te monteren is nog een extra montageplaat ontworpen. Deze heeft een specifieke dikte zodat, mits toevoeging van een profiel van 45 mm, de zwenkwielen met steunpoot hiernaast gemonteerd kunnen worden.

Om het platform nog volledig af te werken dienen er nog gaten geboord te worden in de panelen. Deze dienen voor het laten passeren van kabels en het monteren van drukknoppen. Er moeten ook gaten in de profielen van het blad gemaakt worden om hier ook kabels te laten passeren.

Hoofdstuk 4

Mechanisch ontwerp en analyse

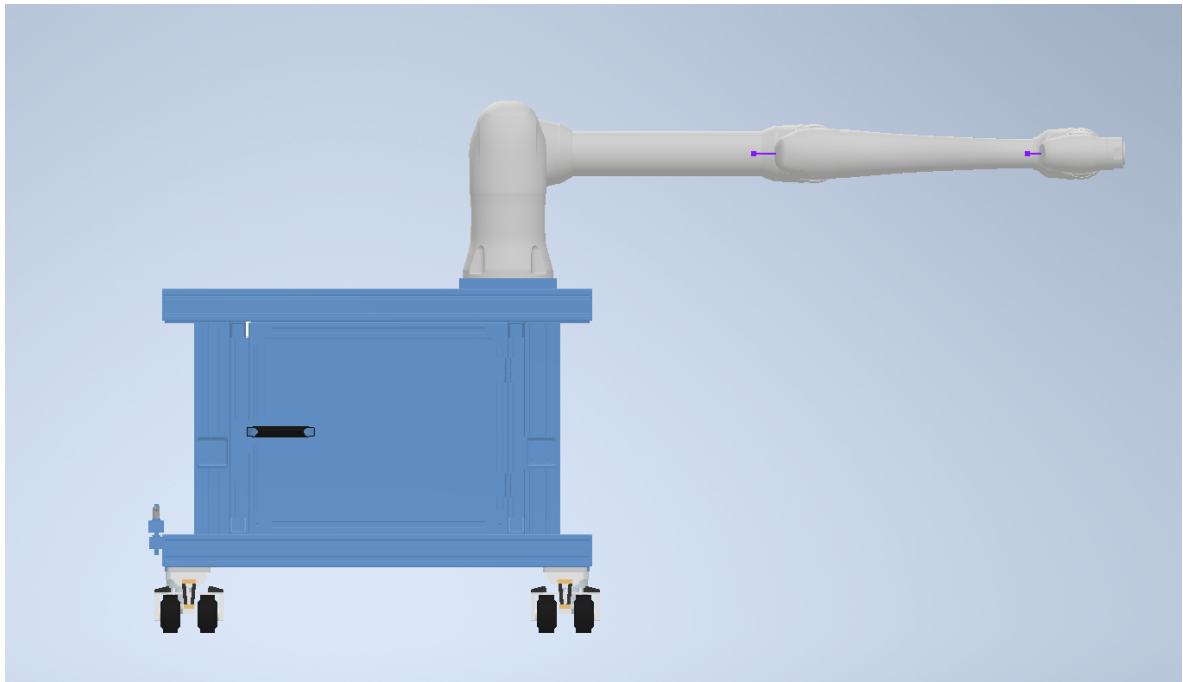
Om een veilige werking en een nauwkeurige uitvoering van de taken te kunnen garanderen, is het noodzakelijk dat het mobiele platform bestand is tegen belastingen die ontstaan tijdens normale werking zonder omver te kantelen. In eerste instantie is er een handmatige berekening uitgevoerd om de stabiliteit en de prestaties te evalueren, vervolgens is deze berekening geautomatiseerd met een Python-script om verdere ontwerpen van het mobiel platform snel en accuraat te kunnen controleren. Dit verhoogt de efficiëntie van het ontwerpproces.

In dit hoofdstuk wordt de analyse aangehaald met de nodige berekeningen om te verifiëren of het minimale gewicht voor bepaalde afmetingen van het mobiele platform overschreden wordt. Om te controleren of de KELO drive 100 een juiste keuze is voor het platform, wordt er ook bekeken of de maximale draagkracht van een wielmodule niet overschreden wordt als gevolg van de belasting. Deze analyse is uitgevoerd voor de transportpositie met statische cobotarm enerzijds, en het meest extreme geval beschreven in de stabiliteitsberekening anderzijds. Omdat nog niet alle componenten juist bekend waren in deze fase, zijn er enkele aannames voor onder andere de afmetingen van het platform gemaakt die ook verderop verduidelijkt worden.

4.1 Situering

Om een correct beeld te verkrijgen, dient de stabiliteitsanalyse uitgevoerd te worden voor het worst-case scenario dat aannemelijk is tijdens normale werking. Figuur 4.1 toont een schets van de situatie, weergegeven in een schermopname van Autodesk Inventor Professional 2025. Het betreft een maximaal belaste, horizontaal uitgestrekte, naarboven accelererende cobotarm gemonteerd op een hoek van het mobiel platform. Verder worden in de berekeningen componenten opgenomen die een significante invloed op de stabiliteit uitoefenen. Voorbeelden hiervan naast het frame en de cobotarm zelf, zijn onder andere de actieve wielmodules, de batterij en de controller voor de cobotarm. Deze componenten zijn op een strategische manier in het frame geplaatst om een zo stabiel mogelijk platform te verkrijgen.

De analyse is uitgevoerd voor een Doosan H2017-cobot. Deze keuze is gebaseerd op het feit dat deze cobotarm de grootste reikwijdte en draagvermogen heeft in vergelijking met de UR5 en KUKA iiwa14. Dit zal resulteren in de zwaarste belasting voor het mobiele platform.



Figuur 4.1: Situatieschets stabiliteitsanalyse

De eerste stap in de berekening is het omzetten van de specificaties van de Doosan H2017 naar een bruikbaar model voor verdere berekeningen. De datasheets beschikbaar gesteld door Doosan Robotics bevatten echter onvoldoende informatie, waardoor enkele aannames noodzakelijk zijn om tot een bruikbaar resultaat te komen.

Na het berekenen van de invloed die de Doosan H2017 heeft op het mobiele platform, worden de massa's van de overige componenten in het platform bepaald. Deze componenten dragen bij aan de stabiliteit door het zwaartepunt (ZP) van het systeem te verlagen en de belasting op de wielmodules gelijkmatiger te verdelen.

In de laatste stap worden alle berekeningen samengevoegd om te verifiëren of het totale gewicht van het platform voldoende is om kantelen te voorkomen en of de wielmodules binnen hun maximale draagkracht blijven tijdens transport. Eenmaal het mobiele platform aan een werkstation gekoppeld staat, zijn er extra mechanische connecties en stabilisatoren aanwezig om beweging te minimaliseren.

4.2 Parameterische stabiliteitsanalyse

Om de berekeningen zo nauwkeurig en realistisch mogelijk uit te voeren, worden de relevante parameters hieronder opgesomd. Deze parameters zijn onderverdeeld in vaste parameters en veranderlijke parameters.

Door deze parameters duidelijk te onderscheiden, kan er efficiënt worden gewerkt met aannames en iteratieve berekeningen om tot een optimaal ontwerp te komen.

4.2.1 Vaste parameters

De vaste parameters omvatten eigenschappen die niet gewijzigd kunnen worden in het ontwerp-proces of die reeds vroegtijdig bepaald zijn. Voorbeelden hiervan zijn de technische specificaties van de KELO-drives, de Doosan H2017-cobotarm en -controller.

Tabel 4.1 vat deze parameters samen. De gegevens zijn bekomen uit datasheets of afgeleid door middel van korte berekeningen van data die te vinden zijn in de voorgenoemde datasheets.

Tabel 4.1: Vaste parameters met betrekking tot de stabiliteitsberekening

Omschrijving	Waarde	Eenheid
Parameters cobot (zie datasheet)	-	-
Afstand tussen de rand van het mobiele platform en de rand van de montageplaat van de cobot	0	mm
Hoogte wielmodules	160	mm
Diameter voetafdruk wielmodules	190	mm
Afstand tussen de rand van het mobiele platform en de rand van een wielmodule	5	mm
Massa wielmodules	5,00	kg
Lengte controller voor cobot	520,00	mm
Breedte controller voor cobot	240,00	mm
Hoogte controller voor cobot	219,00	mm
Massa controller voor cobot	10,00	kg

In tabel 4.1 zijn ook enkele afstanden gegeven relatief tot de rand van het mobiele platform. Deze zijn op voorhand reeds zo gedefinieerd aangezien deze ongeacht de grootte van het platform, aan de rand moeten gemonteerd worden. Dit zal in tabel 4.2 ook herhaald worden.

4.2.2 Variabele parameters met betrekking tot het mobiele platform

Variabele parameters zijn afhankelijk van ontwerpkeuzes en zijn veranderd tussen de verschillende versies. Tabel 4.2 toont die parameters alsook hun waarden bij de berekening van het finale ontwerp. Dit zijn voornamelijk de afmetingen en het gewicht van het frame van extrusieprofielen.

Tabel 4.2: Variabele parameters met betrekking tot de stabiliteitsberekening

Omschrijving	Waarde	Eenheid
Buitenmaat lengte frame	1170,00	mm
Buitenmaat breedte frame	1170,00	mm
Buitenmaat hoogte frame	760,0)	mm
Massa frame	181,30	kg
Lengte batterij	520	mm
Breedte batterij	240	mm
Hoogte batterij	219	mm
Massa batterij	20,50	kg
Afstand tussen de zijkant van de batterij en de rand van het mobiele platform	405,00	mm

4.2.3 Gebruikte methode

De gebruikte methode is gebaseerd op een vergelijkbare analyse reeds uitgevoerd op ACRO voor twee UR20-cobotarmen [39]. Deze berekening wordt uitgevoerd in het XY-vlak.

Aangezien de stabiliteitsberekeningen mee het ontwerp van het mobiele platform sturen, dienden deze voor elke iteratie van het platform opnieuw uitgevoerd te worden. Aangezien dit manueel een tijdrovende taak zou zijn, is er gekozen om dit te automatiseren door middel van een Python-script¹. Hierdoor kan de ontwerper snel feedback krijgen over de stabiliteit van zijn ontwerp.

Traagheidsmomenten

Om verdere berekeningen te vereenvoudigen, worden de massatraagheidsmomenten van de cobotarm en payload voor de positie afgebeeld in figuren 4.1 en 4.3 apart berekend. Indien de configuratie van de cobotarm gewijzigd wordt, bijvoorbeeld door het ellebooggewricht onder een hoek te zetten, dienen deze herberekend te worden. Verder wordt er gebruikt gemaakt van de formule voor de massatraagheid van een puntmassa te berekenen. Deze vereenvoudiging volgt uit het feit dat de exacte locaties van de zwaartepunten van de verschillende componenten van de cobotarm, gegeven zijn in de datasheet voor kinematische en dynamische berekeningen van de Doosan H2017 cobot. Deze datasheet is terug te vinden in bijlage A. Ook zou het correct modelleren van deze componenten uiterst complex zijn aangezien deze doorgaans geen consistente gewichtsverdeling hebben doorheen hun lengte. Aangezien alle componenten eenzelfde rotatieas hebben, namelijk de as door het schoudergewricht, mogen deze eenvoudig opgeteld worden.

$$\begin{aligned}
 I_{cob,a} &= \sum_{comp} I_{comp} \\
 &\Leftrightarrow = \sum_{comp} (m_{comp} \cdot (d_{comp,axis})^2) \\
 &\Leftrightarrow = m_{w2} \cdot (d_{p2,z} - d_{l1})^2 + m_{w3} \cdot (d_{p3,z} - d_{l1})^2 + m_{w4} \cdot (d_{p4,z} - d_{l1})^2 \\
 &\quad + m_{w5} \cdot (d_{p5,z} - d_{l1})^2 + m_{w6} \cdot (d_{p6,z} - d_{l1})^2
 \end{aligned} \tag{4.1}$$

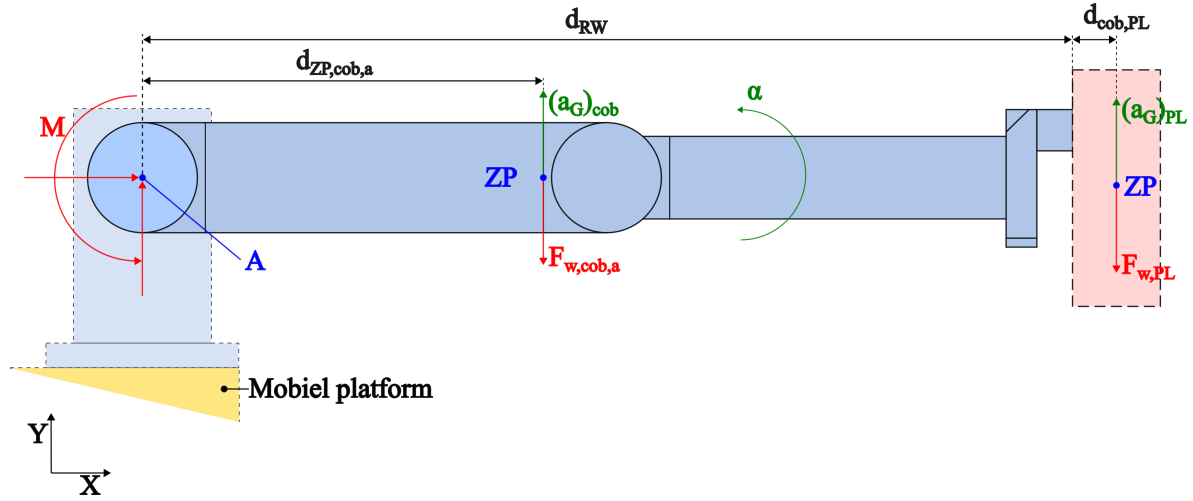
$$I_{cob,pl} = m_{pl} \cdot (d_{RW} + d_{pl})^2 \tag{4.2}$$

De variabelen gebruikt in vergelijking 4.1 zijn samen met een verduidelijkende figuur terug te vinden in bijlage A. Uit vergelijking 4.2 kan afgeleid worden dat het massatraagheidsmoment beïnvloed wordt door de massa van de component, alsook door de afstand tot de rotatieas. Hieruit volgt dat een wijziging in de configuratie van de cobotarm, ook zal resulteren in een wijziging van het massatraagheidsmoment.

Om deze berekening te verifiëren, is er een poging gedaan om het massatraagheidsmoment te berekenen met behulp van Autodesk Inventor. Er is echter geen accuraat model beschikbaar van de cobot wat resulteerde in onrealistische waarden.

¹Beschikbaar op https://gitlab.kuleuven.be/ACRO/masters-theses/mp2425_bram_stijn

Maximale hoekversnelling



Figuur 4.2: Vrijlichaamsschema acceleratie cobotarm

Op basis van het vrijlichaamsschema in figuur 4.2, kan er met de momentenstelling rond de as door het schoudergewricht (A) de maximale hoekversnelling (α) waarmee de cobotarm naar boven kan accelereren met het schoudergewricht, berekend worden. De betekenis, waarde en eenheden van de verschillende variabelen kan teruggevonden worden in tabel 4.3.

$$\begin{aligned}
 \sum M_A &= \sum (M_k)_A \\
 \Leftrightarrow M - F_{w,cob,a} \cdot d_{ZP,cob,a} - F_{w,PL} \cdot (d_{RW} + d_{cob,PL}) &= I_{cob,a} \cdot \alpha + I_{cob,PL} \cdot \alpha \\
 \Leftrightarrow M - m_{cob,a} \cdot g \cdot d_{ZP,cob,a} - m_{PL} \cdot g \cdot (d_{RW} + d_{cob,PL}) &= (I_{cob,a} + I_{cob,PL}) \cdot \alpha \\
 \Leftrightarrow \alpha &= \frac{M - (m_{cob,a} \cdot d_{ZP,cob,a} + m_{PL} \cdot (d_{RW} + d_{cob,PL})) \cdot g}{I_{cob,a} + I_{cob,PL}}
 \end{aligned} \tag{4.3}$$

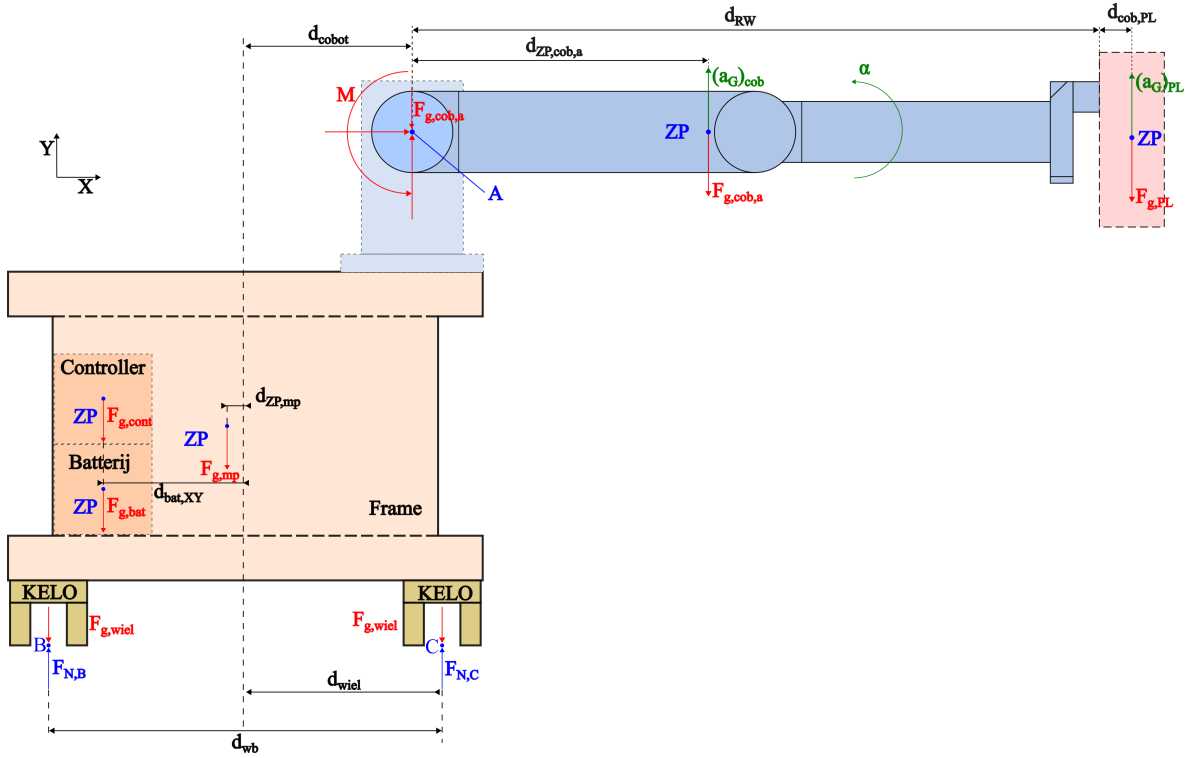
Tabel 4.3: Variabelen en waarden voor het berekenen van de maximale hoekversnelling

Symbool	Omschrijving	Waarde	Eenheid
M	Koppel schoudergewricht	750*	Nm
$m_{cob,a}$	Massa cobotarm (zonder basis)	47,01	kg
$I_{cob,a}$	Traagheidsmoment cobotarm	35,12	kgm ²
$d_{ZP,cob,a}$	Afstand ZP cobot t.o.v. draaias	731,89	mm
m_{pl}	Maximale massa payload (grijper en last)	20,0	kg
$I_{cob,pl}$	Traagheidsmoment payload (grijper en last)	63,37	kgm ²
d_{RW}	Reikwijdte cobot	1700	mm
α	Maximale hoekversnelling cobotarm en payload	0,64	rad/s ²

*Deze waarde is gebaseerd op een vergelijkbare cobot, namelijk een UR20, aangezien deze niet terug te vinden is in datasheets van de Doosan H2017 cobot. Doosan Robotics wenste geen antwoord te geven op de vraag wat de exacte waarde voor de H2017 is.

Minimale massa van het mobiele platform

Figuur 4.3 toont een vereenvoudigde versie van het vrijlichaamsschema van de mobiele manipulator. Om een overzichtelijk schema te behouden, zijn niet alle afstanden die in de berekeningen gebruikt worden, op deze figuur aangebracht. Deze zijn echter wel verder verduidelijkt in tabel 4.4. Met behulp van dit schema kan een stelsel van twee vergelijkingen en twee onbekenden opgesteld en opgelost worden. Vergelijking 4.4 representeert de som van alle krachten in de verticale richting terwijl vergelijking 4.5 de momentenvergelijking rond de as door het schoudergewricht van de cobot voorstelt. Het oplossen van dit stelsel zal als resultaat de kracht verdeeld over de twee wielen aan de rechterkant ($F_{N,C}$) en het minimale gewicht ($F_{g,mp}$) van het mobiele platform geven.



Figuur 4.3: Vrijlichaamsschema van het mobiele platform met de cobotarm in het XY-vlak

$$\begin{aligned}
 \sum F_y &= \sum (m_a \cdot a_{g,y}) \\
 &\Leftrightarrow F_{N,B} + F_{N,C} - F_{g,mp} - 4 \cdot F_{g,wiel} - F_{g,bat} - F_{g,cont} - (F_{g,cob,b} + F_{g,cob,a} + F_{g,pl}) \\
 &= -m_{cob,a} \cdot a_G - m_{cob,pl} \cdot a_G \\
 &\Leftrightarrow F_{N,B} + F_{N,C} - [m_{mp} + 4 \cdot m_{wiel} + m_{bat} + m_{cont} + (m_{cob,b} + m_{cob,a} + m_{pl})] \cdot g \\
 &= [m_{cob,a} \cdot d_{ZP,cob,a} + m_{cob,pl} \cdot (d_{RW} + d_{ZP,cob,pl})] \cdot \alpha
 \end{aligned} \tag{4.4}$$

$$\begin{aligned}
\sum M_A &= \sum (M_k)_A \\
&\Leftrightarrow F_{N,C} \cdot (-d_{cobot} + d_{wiel}) + F_{N,B} \cdot (-d_{cobot} - d_{wiel}) - F_{w,mp} \cdot (-d_{cobot}) \\
&\quad - n_w \cdot F_{g,wiel} \cdot (-d_{cobot} + d_{wiel}) - n_w \cdot F_{g,wiel} \cdot (-d_{cobot} - d_{wiel}) \\
&\quad - (F_{g,bat} + F_{g,cont}) \cdot (-d_{cobot} - d_{bat,XY}) - F_{g,cob,a} \cdot d_{ZP,cob,a} - F_{g,PL} \cdot (d_{RW} + d_{ZP,PL}) \\
&= (I_{cob,a} + I_{cob,pl}) \cdot \alpha \\
&\Leftrightarrow F_{N,C} \cdot (-d_{cobot} + d_{wiel}) + F_{N,B} \cdot (-d_{cobot} - d_{wiel}) - [m_{mp} \cdot (-d_{cobot}) \\
&\quad + n_w \cdot m_{wiel} \cdot (-d_{cobot} + d_{wiel}) + n_w \cdot m_{wiel} \cdot (-d_{cobot} - d_{wiel}) \\
&\quad + (m_{bat} + m_{cont}) \cdot (-d_{cobot} - d_{bat,XY}) + m_{cob,a} \cdot d_{ZP,cob,a} + m_{PL} \cdot (d_{RW} + d_{ZP,PL})] \cdot g \\
&= (I_{cob,a} + I_{cob,pl}) \cdot \alpha
\end{aligned} \tag{4.5}$$

Tabel 4.4: Variabelen en waarden voor het berekenen van het minimale platformgewicht

Symbol	Omschrijving	Waarde	Eenheid
$m_{cob,a}$	Massa van de cobotarm (zonder basis)	47,01	kg
$m_{cob,b}$	Massa van de basis van de cobot	20,25	kg
d_{cobot}	Afstand tussen de as door het schoudergewricht van de cobot en het middenvlak van het mobiele platform	450	mm
$I_{cob,a}$	Traagheidsmoment van de cobotarm	35,12	kgm ²
$d_{ZP,cob,a}$	Afstand tussen het zwaartepunt van de cobotarm en de as door het schoudergewricht van de cobot	731,89	mm
m_{pl}	Maximale massa van de payload (grijper en last)	20	kg
$I_{cob,pl}$	Traagheidsmoment van de payload (grijper en last)	63,37	kgm ²
$d_{ZP,mp}$	Afstand tussen het zwaartepunt van het frame en het middenvlak van het mobiel platform (verondersteld perfect in het midden)	0	mm
d_{RW}	Reikwijdte van de cobot	1700	mm
m_{wiel}	Massa van een wielmodule	5	kg
d_{wiel}	Afstand tussen de hartlijn van de wielmodules en het middenvlak van het mobiele platform	485	mm
n_w	Aantal wielen per zijde van het mobiele platform	2	-
$m_{bat,XY}$	Massa van de batterij	20,25	kg
m_{cont}	Massa van de controller voor de cobot	10,00	kg
d_{bat}	Afstand tussen de batterij en de controller die zich er boven bevindt en het middenvlak van het mobiele platform	285,00	mm
$F_{N,B}$	Normaalkracht verdeeld over de linkse wielen (indien deze wielen geen kracht meer ondervinden, zal het platform omkantelen)	0	N
$F_{N,C}$	Normaalkracht verdeeld over de rechtse wielen	2122,99	N
m_{mp}	Minimaal gewicht van het frame	83,22	kg

4.2.4 Resultaten

De berekende massatraagheid van het bewegende deel van de cobotarm bedraagt $I_{cob,a} = 35,52 \text{ kgm}^2$. Verder bedraagt deze voor de maximale payload van $m_{pl} = 20 \text{ kg}$ op de maximale afstand van het einde van de cobotarm $I_{cob,pl} = 63,37 \text{ kgm}^2$. Dit geeft samen een totale massatraagheid van $I_{cob} = 98,79 \text{ kgm}^2$ voor deze horizontale configuratie.

De maximale hoekversnelling die mogelijk is door te versnellen met het schoudergewricht waarvan verondersteld wordt dat deze een koppel van $M = 750 \text{ Nm}$ kan leveren, is $\alpha = 0,64 \text{ rad/s}^2$ oftewel $36,67^\circ/\text{s}^2$.

Om de laatste iteratie van het platform stabiel te houden, is een massa van $m_{mp} = 83,22 \text{ kg}$ vereist voor het frame van extrusieprofielen. Volgens de MachineBuilder van Vention bedraagt het gewicht van het dit frame $181,30 \text{ kg}$ en is het dus voldoende om het platform stabiel te houden tijdens normale werking.

Een belangrijke opmerking bij deze resultaten is dat de berekeningen uitgevoerd zijn voor maximale waarden die tijdens normaal bedrijf verwacht worden. Het is mogelijk dat deze overschreden worden in een noodsituatie met een noodstop van de cobotarm. Hierdoor hoort de cobotarm tijdens verplaatsingen stationair te zijn. Wanneer het mobiele platform aan een werkstation gekoppeld wordt, zal de bijkomende mechanische connectie ervoor zorgen dat het onmogelijk is voor het platform om om te kantelen in deze extreme situaties, moesten die zich voordoen.

4.3 Belastingsanalyse

4.3.1 Krachten op actieve wielmodules

XY-vlak: links en rechts

Door het effectieve gewicht ($m_{mp} = 181,30 \text{ kg}$) van het mobiele platform te implementeren in formules 4.4 en 4.5, kan het gewicht verdeeld over de twee rechtse ($F_{N,R}$) en twee linkse ($F_{N,L}$) wielmodulus gevonden worden.

$$\begin{aligned}
 \sum F_y &= \sum (m_a \cdot a_{g,y}) \\
 &\Leftrightarrow F_{N,B} + F_{N,C} - F_{g,mp} - 4 \cdot F_{g,wiel} - F_{g,bat} - F_{g,cont} - (F_{g,cob,b} + F_{g,cob,a} + F_{g,pl}) \\
 &= m_{cob,a} \cdot a_G - m_{cob,pl} \cdot a_G \\
 &\Leftrightarrow F_{N,B} + F_{N,C} - [m_{mp} + 4 \cdot m_{wiel} + m_{bat} + m_{cont} + (m_{cob,b} + m_{cob,a} + m_{pl})] \cdot g \\
 &= [m_{cob,a} \cdot d_{ZP,cob,a} + m_{cob,pl} \cdot (d_{RW} + d_{ZP,cob,pl})] \cdot \alpha
 \end{aligned} \tag{4.6}$$

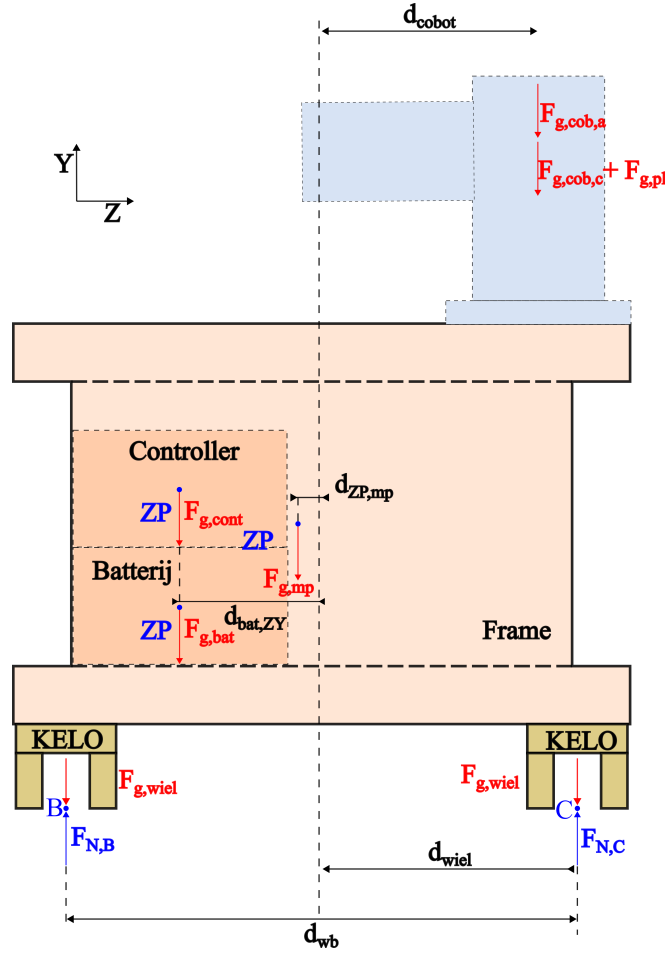
$$\begin{aligned}
\sum M_A &= \sum (M_k)_A \\
&\Leftrightarrow F_{N,R} \cdot (-d_{cobot} + d_{wiel}) + F_{N,L} \cdot (-d_{cobot} - d_{wiel}) - F_{w,mp} \cdot (-d_{cobot}) \\
&\quad - n_w \cdot F_{g,wiel} \cdot (-d_{cobot} + d_{wiel}) - n_w \cdot F_{g,wiel} \cdot (-d_{cobot} - d_{wiel}) \\
&\quad - (F_{g,bat} + F_{g,cont}) \cdot (-d_{cobot} - d_{bat,XY}) - F_{g,cob,a} \cdot d_{ZP,cob,a} - F_{g,PL} \cdot (d_{RW} + d_{ZP,PL}) \\
&= (I_{cob,a} + I_{cob,pl}) \cdot \alpha \\
&\Leftrightarrow F_{N,R} \cdot (-d_{cobot} + d_{wiel}) + F_{N,L} \cdot (-d_{cobot} - d_{wiel}) - [m_{mp} \cdot (-d_{cobot}) \\
&\quad + n_w \cdot m_{wiel} \cdot (-d_{cobot} + d_{wiel}) + n_w \cdot m_{wiel} \cdot (-d_{cobot} - d_{wiel}) \\
&\quad + (m_{bat} + m_{cont}) \cdot (-d_{cobot} - d_{bat,XY}) + m_{cob,a} \cdot d_{ZP,cob,a} + m_{PL} \cdot (d_{RW} + d_{ZP,PL})] \cdot g \\
&= (I_{cob,a} + I_{cob,pl}) \cdot \alpha
\end{aligned} \tag{4.7}$$

Tabel 4.5: Variabelen en waarden voor het berekenen van de reactiekrachten op de linkse en rechtse wielmodules

Symbool	Omschrijving	Waarde	Eenheid
$m_{cob,a}$	Massa van de cobotarm (zonder basis)	47,01	kg
$m_{cob,b}$	Massa van de basis van de cobot	20,25	kg
d_{cobot}	Afstand tussen de as door het schoudergewricht van de cobot en het middenvlak van het mobiele platform	450	mm
$I_{cob,a}$	Traagheidsmoment van de cobotarm	35,12	kgm ²
$d_{ZP,cob,a}$	Afstand tussen het zwaartepunt van de cobotarm en de as door het schoudergewricht van de cobot	731,89	mm
m_{pl}	Maximale massa van de payload (grijper en last)	20	kg
$I_{cob,pl}$	Traagheidsmoment van de payload (grijper en last)	63,37	kgm ²
d_{RW}	Reikwijdte van de cobot	1700	mm
m_{wiel}	Massa van een wielmodule	5	kg
d_{wiel}	Afstand tussen de hartlijn van de wielmodules en het middenvlak van het mobiele platform	485	mm
n_w	Aantal wielen per zijde van het mobiele platform	2	-
$m_{bat,XY}$	Massa van de batterij	20,25	kg
m_{cont}	Massa van de controller voor de cobot	10,00	kg
d_{bat}	Afstand tussen de batterij en de controller die zich er boven bevindt en het middenvlak van het mobiele platform	285,00	mm
m_{mp}	Gewicht van het Vention-frame	181,30	kg
$F_{N,L}$	Normaalkracht verdeeld over de linkse wielen	481,01	N
$F_{N,R}$	Normaalkracht verdeeld over de rechtse wielen	2604,22	N

ZY-vlak: voor en achter

Eenzelfde berekening is mogelijk in het vlak 90° gedraaid met het XY-vlak, namelijk het ZY-vlak. Door de oriëntatie van de cobot, zal de acceleratie van de cobot hier geen invloed hebben en kan er statisch gerekend worden. Door het vierkante ontwerp van het mobiele platform, blijven veel waarden hetzelfde tussen deze twee berekeningen en zijn de wijzigingen aan het vrijlichaamsschema beperkt, zoals zichtbaar in figuur 4.4. Analooq aan voorgaande berekeningen stelt vergelijking 4.8 een krachtsommatie in de Y-richting voor, en vergelijking 4.9 een momentenstelling. Hierbij is as D gedefinieerd als een as die 90° gedraaid is ten opzichte van as A, maar wel dezelfde afstand heeft tot het middenvlak van het mobiele platform. De waarden die ingevuld dienen te worden, zijn te vinden in tabel 4.6.



Figuur 4.4: Vrijlichaamsschema van het mobiele platform met de cobotarm in het ZY-vlak

$$\begin{aligned}
 \sum F_y &= 0 \\
 \Leftrightarrow F_{N,V} + F_{N,A} - F_{g,mp} - 4 \cdot F_{g,wiel} - F_{g,bat} - F_{g,cont} - (F_{g,cob,b} + F_{g,cob,a} + F_{g,pl}) &= 0 \\
 \Leftrightarrow F_{N,V} + F_{N,A} - [m_{mp} + 4 \cdot m_{wiel} + m_{bat} + m_{cont} + (m_{cob,b} + m_{cob,a} + m_{pl})] \cdot g &= 0
 \end{aligned} \tag{4.8}$$

$$\begin{aligned}
& \sum M_D = 0 \\
& \Leftrightarrow F_{N,V} \cdot (-d_{cobot} + d_{wiel}) + F_{N,A} \cdot (-d_{cobot} - d_{wiel}) - F_{w,mp} \cdot (-d_{cobot}) \\
& \quad - n_w \cdot F_{g,wiel} \cdot (-d_{cobot} + d_{wiel}) - n_w \cdot F_{g,wiel} \cdot (-d_{cobot} - d_{wiel}) \\
& \quad - (F_{g,bat} + F_{g,cont}) \cdot (-d_{cobot} - d_{bat,ZY}) = 0 \\
& \Leftrightarrow F_{N,R} \cdot (-d_{cobot} + d_{wiel}) + F_{N,L} \cdot (-d_{cobot} - d_{wiel}) - [m_{mp} \cdot (-d_{cobot}) \\
& \quad + n_w \cdot m_{wiel} \cdot (-d_{cobot} + d_{wiel}) + n_w \cdot m_{wiel} \cdot (-d_{cobot} - d_{wiel}) \\
& \quad + (m_{bat} + m_{cont}) \cdot (-d_{cobot} - d_{bat,ZY})] \cdot g = 0
\end{aligned} \tag{4.9}$$

Tabel 4.6: Variabelen en waarden voor het berekenen van de krachten verdeeld over de voorste en achterste wielmodules

Symbol	Omschrijving	Waarde	Eenheid
$m_{cob,a}$	Massa van de cobotarm (zonder basis)	47,01	kg
$m_{cob,b}$	Massa van de basis van de cobot	20,25	kg
m_{pl}	Maximale massa van de payload (grijper en last)	20	kg
d_{cobot}	Afstand tussen de as loodrecht op de as door het schoudergewricht van de cobot en het middenvlak van het mobiele platform	450	mm
m_{wiel}	Massa van een wielmodule	5	kg
d_{wiel}	Afstand tussen het raakvlak van de wielmodules en het middenvlak van het mobiele platform	485	mm
n_w	Aantal wielen per zijde van het mobiele platform	2	-
m_{bat}	Massa van de batterij	20,25	kg
m_{cont}	Massa van de controller voor de cobot	10,00	kg
$d_{bat,ZY}$	Afstand tussen batterij en de controller die zich er boven bevindt en het middenvlak van het mobiele platform	145,00	mm
m_{mp}	Gewicht van het Vention-frame	181,30	kg
$F_{N,V}$	Normaalkracht verdeeld over de voorste wielen	1880,56	N
$F_{N,A}$	Normaalkracht verdeeld over de achterste wielen	1204,67	N

Combineren van de resultaten

Nu de reactiekrachten aan de verschillende zijden van het mobiele platform berekend zijn, worden deze gecombineerd om een indicatieve waarde te krijgen voor elk hoekpunt, waar een wielmodule gemonteerd is.

Uit tabel 4.6 kan een procentuele gewichtsverdeling opgesteld worden voor de voor- en achterkant van de mobiele manipulator. Deze kan dan toegepast worden op de resultaten zichtbaar in tabel 4.5 om de lasten voor de individuele wielmodules te bekomen.

Tabel 4.7: Lasten voor de individuele wielmodules

Omschrijving	Waarde	Eenheid
Procentuele gewichtsverdeling voorkant	60,95	%
Procentuele gewichtsverdeling achterkant	39,05	%
Massa op wielmodule rechtsvoor	161,81	kg
Massa op wielmodule linksvoor	29,89	kg
Massa op wielmodule rechtsachter	103, 65	kg
Massa op wielmodule linksachter	19,15	kg

Tabel 4.7 toont de berekende belasting voor elke wielmodule. Als gevolg van de uitstekende cobotarm, wordt de rechterkant van het platform significant zwaarder belast dan de linkerkant. Bovendien wordt de wielmodule die zich recht onder de cobotarm bevindt, in de extreme situatie, overbelast aangezien de maximale capaciteit 125 kg bedraagt. Om te voorkomen dat deze wielmodule beschadigd wordt, is het aanbrengen en gebruiken van een extra steun noodzakelijk bij het gebruik van de cobot.

Belangrijker echter, is dat er geen wielmodules overbelast worden tijdens transport tussen verschillende werkstations. Om dit te verzekeren, dient de cobot in een transportpositie geplaatst te worden. Deze transportpositie moet resulteren in een betere gewichtsverdeling en voornamelijk de wielmodule onder de cobot ontlasten.

4.3.2 Structurele integriteit

Na het doorlopen van verschillende ontwerpen, is het voorlaatste ontwerp voorgesteld aan twee ingenieurs werkzaam bij Vention. Op basis van de kennis en ervaring van deze ontwerpers, is er gekozen om extrusieprofielen met grotere dimensies te verwerken in het platform. Hierdoor is het ontwerp gebaseerd op (45 x 45)mm extrusieprofielen aangepast naar een dat (90 x 90)mm toepast. Deze grotere profielen zijn vervaardigd uit hetzelfde materiaal maar hebben door hun grotere afmetingen een hogere buigsterkte wat het gehele platform verstevigen. Het gewicht van 181,30 kg doorheen het hoofdstuk is ook op deze herwerkte versie gebaseerd.

Hoofdstuk 5

Elektrisch ontwerp

In dit hoofdstuk wordt de selectie van de elektrische componenten geduid. Verder wordt er ook een risicoanalyse gemaakt van het mobiele platform en wordt er een veiligheidsconcept opgesteld.

5.1 Indeling

Bij het mechanisch ontwerp is reeds rekening gehouden met aparte compartimenten voor componenten op wisselspanning en vermogensverdeling binnen het mobiele platform, en voor componenten op een lage gelijkspanning die instaan voor de controle van het platform en dataoverdracht. Deze scheiding biedt als voordeel dat de gevoelige componenten verwijderd worden van componenten die mogelijks elektromagnetische velden genereren en op die manier dataoverdracht verstoren. Verder worden mogelijke gevolgen van storingen beperkt in hun ernst.

5.1.1 Vermogenzijde

Aan de vermogenzijde zijn volgende componenten terug te vinden:

- batterijlader;
- batterij;
- controller voor de cobot;
- hoofdschakelaar om de installatie spanningsloos te stellen;
- klemmenstrook om stroomverdeling te faciliteren.

5.1.2 Stuurzijde

Aan de stuurzijde zijn volgende componenten terug te vinden:

- industriële computer;
- EK1100 IO-eiland;
- EtherCAT-hub voor connectie met de KELO 100 drives;
- Sick veiligheidscontroller;
- klemmenstrook om doorverbindingen te maken.

5.1.3 Overige

5.2 Risicoanalyse

Om de benodigde besturingstechnische componenten te bepalen, werd er een risicoanalyse uitgevoerd op het ontworpen platform. Deze risicoanalyse is ook essentieel om een CE-markering te verkrijgen indien een versie van dit project in de toekomst commercieel beschikbaar gemaakt zou worden. Deze markering toont aan dat de ontworpen en gebouwde machine voldoet aan de zogenoemde Europese machinerichtlijn 2006/42/EG. Voor deze analyse wordt de cobot buiten beschouwing gelaten. Deze behoort niet tot het zelfontworpen gedeelte van het mobiele platform en is volgens de definitie van een cobot al veilig.

De voorgenoemde richtlijn harmoniseert een verscheidenheid aan normen maar concreet worden deze in drie categorieën opgedeeld:

1. A-normen: algemene normen die van toepassing zijn op alle machines.
2. B-normen: specifieke veiligheidsmiddelen en ergonomische aspecten.
3. C-normen: specifieke typemachines.

De risicoanalyse die hieronder beschreven staat, valt onder de categorie A-normen. Deze is uitgevoerd door de norm EN ISO 14121-1 te volgen aangezien deze de de facto standaard is voor risicoanalyses.

5.2.1 Ruimtelijke grenzen van de machine

In dit eerste onderdeel van de analyse is het van belang dat er een duidelijke omschrijving wordt gegeven van de functionele mogelijkheden van de machine. Dit wordt opgedeeld in de volgende punten:

Locatie

De machine bevindt zich in een afgesloten werkplaats en is hierdoor beschermd tegen diverse externe invloeden zoals regen, wind en zonlicht.

Onderdelen

De machine bestaat uit verschillende onderdelen. De voornaamste zijn:

- aandrijfwielen (4 stuks KELO drive 100),
- 24VDC batterij,
- elektrische verdeelinstallatie (exclusief veiligheidscomponenten),
- controlerende PC,
- LiDAR-sensoren,
- frame van extrusieprofielen,
- vergrendelmechanisme voor stationaire stabilisatie.

Beoogd gebruik van de machine

Het mobiel platform is ontworpen voor gebruik in een industriële omgeving. Het dient manueel verplaatst te worden tussen verschillende werkstations. Tijdens de verplaatsing assisteren de aandrijfwielen de operator bij het in beweging zetten en houden van het platform. Aan de werkstations kan de cobot dan taken uitvoeren zoals degene voorgesteld door Opsinox in sectie 2.9.

Bewegingsruimte voor operatoren

Tijdens normale werking (taak uitvoeren) moet er niemand aanwezig zijn in de buurt van het mobiel platform en cobot. Tijdens de verplaatsing is er voldoende ruimte aanwezig rond het platform om vrij te bewegen.

Wanneer een koppeling aan een werkstation vereist is, moet er een operator bij komen. Op andere momenten is de aanwezigheid van personen in de nabijheid van het mobiel platform en de cobotarm niet vereist.

Personen buiten de operator

Tijdens normaal gebruik worden enkel operatoren aan de machine blootgesteld. Voor eventuele storingen op te lossen kunnen ook onderhoudstechniekers aan de gevarenzone blootgesteld worden. Andere personen kunnen wel in de nabijheid aanwezig zijn.

5.2.2 Gebruikersgrenzen

Gebruikers

Het mobiele platform dient bediend te worden door volwassen operatoren met een normaal reactievermogen. Aan de installatie worden geen minderjarige of mindervalide personen tewerkgesteld. Onderhoud en interventies worden door daarvoor opgeleide personen uitgevoerd.

Opleiding

Personen die worden blootgesteld aan het mobiele platform dienen een korte opleiding te volgen om vertrouwd te raken met de bedieningsorganen en het gedrag van het platform alsook de manuele acties die rondom het platform dienen uitgevoerd te worden. Deze omvatten het maken van zowel de elektrische als de mechanische koppeling van de mobiele manipulator aan het werkstation.

Kennis van de operatoren

Aangezien het een nieuwe machine betreft, dienen de operatoren nog inzicht te verwerven in het gebruik van de machine alsook de gevaren die zouden kunnen optreden bij het normaal gebruik van de machine.

Gebruik

De machine wordt enkel overdag onder kunstmatige verlichting gebruikt. De gebruiksfasen van de machine zijn als volgt:

- opbouw,
- onderhoud en
- productie.

Ongevallengeschiedenis

Aangezien het mobiele platform een prototype is, is er nog geen informatie beschikbaar over reeds gebeurde ongevallen.

5.2.3 Latente gevaren

De tweede stap van de risicoanalyse omvat het identificeren van alle latente gevaren, gevaarlijke situaties en gevaarlijke gebeurtenissen die met een machine in verband gebracht worden. Enkele worden in de volgende alinea's beschreven.

Het frame van het mobiel platform bestaat uit afgezaagde extrusieprofielen. Afhankelijk van de graad van afwerking waarin deze geleverd worden, bestaat het risico dat de techniek die de montage uitvoert, met scherpe randen moet werken. Verder kunnen er ook op het afgemonteerde platform scherpe randen zijn waar personen zich aan kunnen bezeren. Door het gebruik van assemblageplaten worden de meest scherpe randen wel afgedekt, maar deze introduceren dan weer oppervlakken waar kledij of andere objecten achter kunnen blijven haken, en meegesleurd worden indien het platform beweegt.

Om het platform te laten bewegen zijn er gemotoriseerde wielmodules gemonteerd onder de hoekpunten. Deze hebben voldoende vermogen om over objecten op hun pad te rijden. Indien een operator bijvoorbeeld zijn voet voor een wiel zou zetten, is de kans reëel dat het platform deze zou overrijden en een kneuzing of breuk veroorzaken.

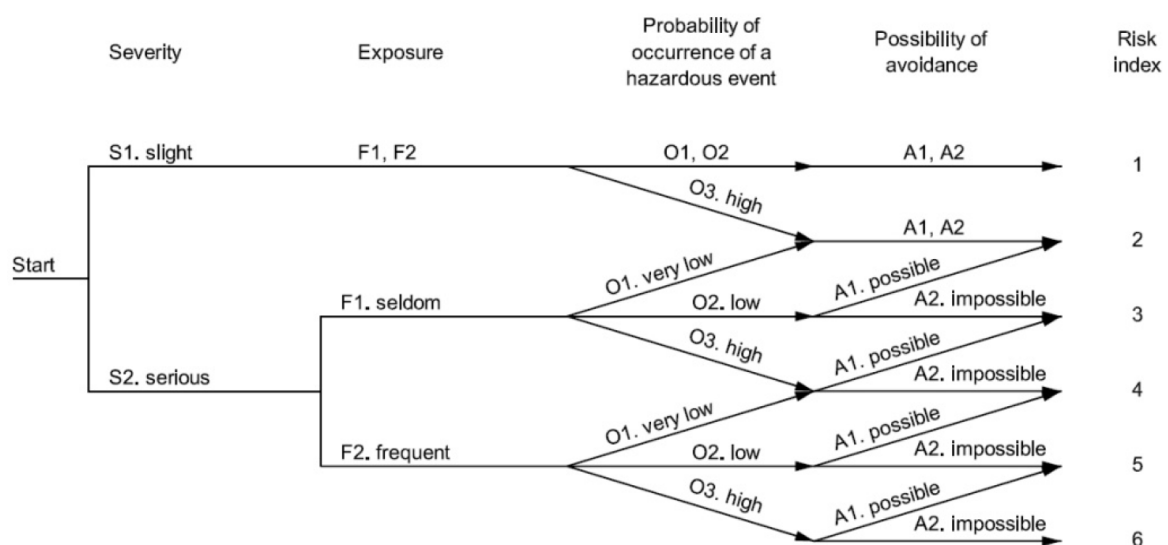
Het merendeel van de elektrische installatie in het mobiele platform bevindt zich op een spanningsniveau van 24 volt gelijkspanning. Hoewel dit in theorie veilig is om aan te raken, kunnen externe factoren zoals natte handen en schoeisel een zodanige invloed hebben dat het onveilig wordt. Verder is er ook 230 volt wisselspanning aanwezig om de batterij bij te laden of om de centrale controller te voeden.

Indien het controlesysteem van het mobiel platform een storing heeft, bestaat de kans dat het platform beweegt zonder een ogenschijnlijke oorzaak. Dit kan als gevolg hebben dat personen gekneld kunnen komen te zitten tussen het mobiel platform en andere objecten in de nabijheid ervan.

Indien een storing zich bevindt in het elektrische systeem, bestaat de kans dat bepaalde componenten en/of bedrading blootgesteld worden aan een te grote spanning of stroom. Dit kan als gevolg hebben dat deze oververhitten en spontaan ontbranden.

5.2.4 Risico-inschatting

De derde stap omvat het inschatten van de risico's veroorzaakt door de latente gevaren. Een risico wordt als acceptabel beschouwd indien zijn risico-index kleiner of gelijk is aan twee. De risico-index van een gevaarlijke situatie wordt bepaald volgens een kwalitatieve methode met een risicograaf. De gebruikte risicograaf is te zien in figuur 5.1.



Figuur 5.1: Risicograaf [40, p. 18]

Volgende paragraaf is geleend van de cursus machineveiligheid opgesteld door PILZ en definieert het gebruik van deze risicograaf:

1. De ernstgraad, de S-factor ('Severity'), de mate van verwonding:
 - S1 = lichte verwonding die omkeerbaar is, bijvoorbeeld een schram of blauwe plek op de ledematen
 - S2 = zware verwonding die onomkeerbaar is, bijvoorbeeld snijwonden, botbreuken, blijvende invaliditeit en beroepsziekten tot zelfs de dood.

2. De blootstelling, de E-factor ('Exposure'), is opgebouwd uit twee parameters, namelijk: de frequentie waaraan de bediener aan het gevaar wordt blootgesteld; de tijdsduur van de bewuste blootstelling. De frequentie van de blootstelling is als volgt gespecificeerd:
 - E1 = 2 maal of minder per shift, of als de totale gecumuleerde blootstelling per shift kleiner is dan 15min
 - E2 = meer dan 2 maal shift, of als de totale gecumuleerde blootstelling per shift groter is dan 15min
3. De waarschijnlijkheid, de O-factor ('Occurence'), voorspelt de kans op een gevaarlijke gebeurtenis:
 - O1 = volwassen, robuuste technologie die voldoende getest is in veiligheidsapplicaties
 - O2 = technische fout gezien tijdens de laatste 2 jaar: Ongepaste menselijke handeling door een persoon met voldoende kennis van de risico's en met meer dan 6 maand werkervaring aan het desbetreffende werkstation.
 - O3 = Technische fout die regelmatig wordt opgemerkt (elke 6 maanden of minder):
 - Ongepaste menselijke handeling door een persoon met minder dan 6 maand werkervaring aan het desbetreffende werkstation
 - Gelijkaardig ongeval voorgekomen in de plant tijdens de laatste 10 jaar

[40, p. 18]

Indien deze risicograaf toegepast wordt op de latente gevaren geïdentificeerd in paragraaf 5.2.3, wordt volgende tabel bekomen:

Tabel 5.1: Risico-inventarisatie van gevaarlijke situaties

Gevaarlijke situatie	Mogelijk gevolg	S	F	O	A	RI
Scherpe randen aan platform	Schaafwonden, sneetjes	1	1	2	1	2
Niet afgeschermdde wielen	Gekneusde tenen	2	2	2	1	4
(In)directe aanraking van elektrische geleiders	Brandwonden, elektrisering, elektrocutie	2	2	2	1	4
Bruuske beweging van het mobiel platform	Kneuzingen	2	2	2	2	5
Omvallen van het mobiel platform	Kneuzingen	2	2	2	2	5
Brand	Brandwonden, rook inademen	2	1	1	1	2

5.2.5 Risicoreductiemaatregelen en evaluatie

Uit de uitgevoerde risicoanalyse blijkt dat de grootste aandachtspunten zijn:

- niet afgeschermd elektrische installatie;
- niet afgeschermd wielen;
- onvoorspelbare bewegingen van het platform.

Vooraleer risico's opgelost worden met logica, is het nodig om te bekijken of deze op een mechanische manier gereduceerd kunnen worden tot een risico-index kleiner dan twee.

Om personen te beschermen tegen toevallige aanrakingen van de elektrische installatie, dient deze afgeschermd te worden met een mechanische barrière die niet-moedwillige aanraking onmogelijk maakt. Indien deze barrière overtreden wordt, dient het mobiele platform gedeactiveerd te worden. De wielen dienen afgeschermd te worden zodanig dat objecten aan de kant geforceerd worden als deze eronder terecht dreigen te komen. Verder is het ook aangeraden om een stootrubber te implementeren rond de buitenzijde van het mobiel platform die een zekere maten van bescherming biedt indien personen aangereden worden door het mobiele platform. Deze aanrijdingen kunnen enkel met behulp van veiligheidslogica vermeden worden. Dit dient dan ook geïmplementeerd te worden. Om kantelen te voorkomen dient het mobiel platform op een mechanische manier bevestigd te worden aan het werkstation. Dit werkstation zal dan extra stabiliteit bieden.

Tabel 5.2: Risico evaluatie na reductiemaatregelen

Gevaarlijke situatie	Mogelijk gevolg	S	F	O	A	RI
Niet afgeschermd wielen	Gekneusde tenen	2	1	1	1	2
(In)directe aanraking van elektrische geleiders	Brandwonden, elektrisering, elektrocutie	2	1	2	1	2
Bruuske beweging van het mobiel platform	Kneuzingen	2	1	1	2	2
Omvallen van het mobiel platform	Kneuzingen	2	1	1	2	2

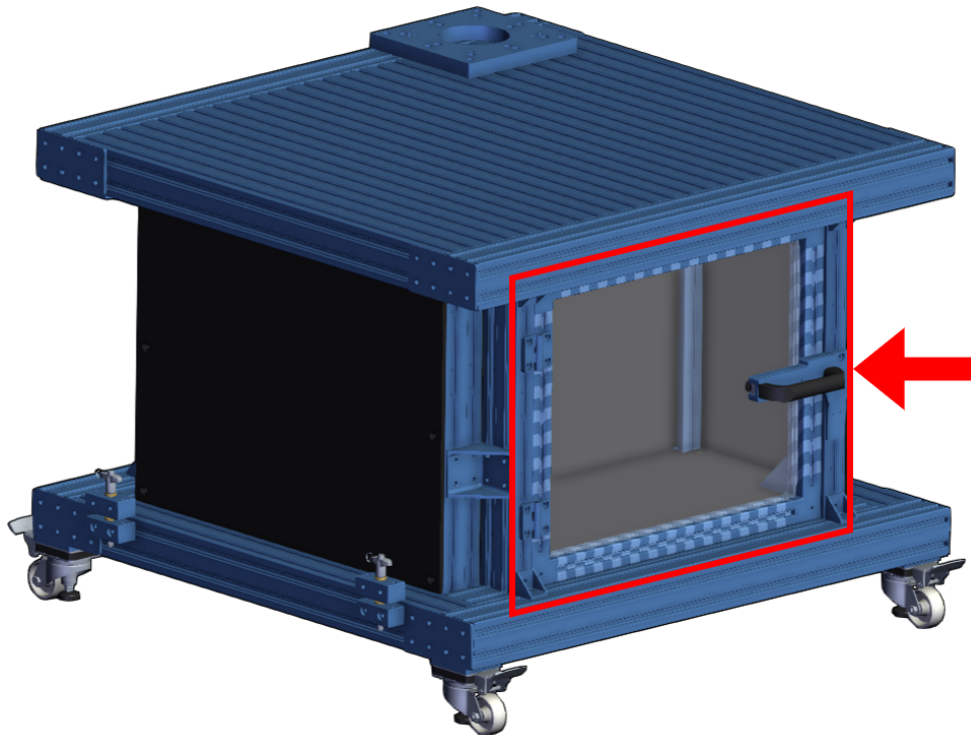
5.3 Veiligheidsconcept

5.3.1 Overzicht en beschrijving

Het doel van het veiligheidsconcept is het veilig maken van de machine maar toch de functionaliteit behouden. Om dit te realiseren zijn in de volgende alinea's acties opgesteld die het mobiel platform veiliger maken.

5.3.2 Uitwerking: mechanische acties

Zoals onder paragraaf 5.2.5 beschreven, is er een mechanische afscherming nodig voor zowel de actieve wielmodules als de elektrische installatie. De afscherming van de wielmodules moet objecten hoger als tien millimeter uit het pad van de wielen kunnen verwijderen. De elektrische installatie wordt gescheiden door middel van plaatmateriaal dat vast gemonteerd is aan het frame van extrusieprofielen. De (de)montage van dit plaatmateriaal is enkel mogelijk met behulp van gereedschap. Om vervanging en/of wijzigingen van componenten toe te laten, zijn er ook twee deuren voorzien (aangeduid met een rode pijl). Deze bestaan uit een kleiner frame van extrusieprofielen waarin een plexiglazen plaat zit en een mechanische grendel. Deze deuren zijn ook enkel te (de)monteren met gereedschap. Bovenop deze mechanische grendel is het aangeraden een (hang)slot te bevestigen om ongeautoriseerde toegang te vermijden. Figuur 5.2 is een visualisatie van deze acties toegepast op het finale ontwerp van Vention. Verder is het ook aangeraden stootrubbers te bevestigen aan de randen van het mobiel platform. Op deze manier kunnen personen beschermd worden tegen de scherpe randen alsook voor een stuk tegen aanrijdingen van het mobiele platform



Figuur 5.2: Vention platform met aangeduide deur

5.3.3 Uitwerking: besturingstechnische acties

Zoals eerder vermeld, moeten alle gevaarlijke situaties die risico's inhouden die niet opgelost kunnen worden op een mechanische manier, vermeden worden via elektrische besturing. Normaal is een veiligheidsrelais de eerste optie om deze sturing op een zo laag mogelijk niveau te realiseren.

Door de mobiele aard van het project, is hier echter gekozen om veiligheidslogica te implementeren met een veiligheidscontroller. Deze methode laat toe rekening te houden met verschillende signalen die op hetzelfde moment ontvangen worden van zowel de magnetische deurcontacten, de noodstopdrukknoppen als de veiligheidslaserscanner. Op basis daarvan stelt de veiligheidscontroller de installatie op een gepaste manier veilig. Enkele voorbeelden van mogelijke ingrepen zijn het volledig stoppen van het proces of het proces verder zetten aan een gereduceerde snelheid. Ook het stopzetten van de wielmodules indien een aanrijding dreigt te gebeuren, is een taak van de veiligheidscontroller.

5.3.4 Uitwerking: diverse acties

De meeste risico's zouden reeds gereduceerd moeten zijn met de mechanische en besturingstechnische acties zoals beschreven in paragrafen 5.3.2 en 5.3.3 respectievelijk.

Indien er tijdens de evolutie van het project, risico's opduiken die niet opgevangen kunnen worden op die manier, kan er nog gekozen worden om specifieke pictogrammen aan te brengen om gebruikers te waarschuwen voor bepaalde gevaren. Aangezien het aanbrengen van deze pictogrammen slechts een minimale kost is, wordt er aangeraden deze aan te brengen. Verder kunnen er ook procedures ontwikkeld worden indien er specifieke interventies zoals bijvoorbeeld onderhoudswerkzaamheden moeten uitgevoerd worden.

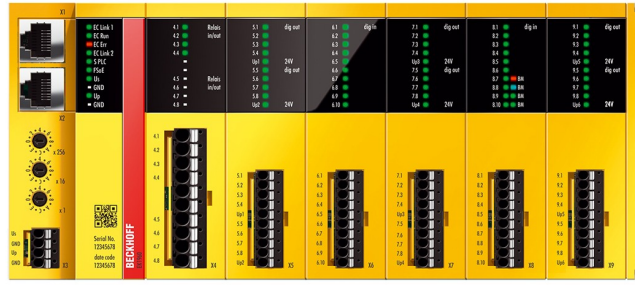
5.4 Componenten

5.4.1 Veiligheid

Tijdens de marktstudie zijn er bij verschillende merken geschikte componenten gevonden. Doordat de ACRO-onderzoeksgroep componenten aan een voordeliger tarief kan aankopen bij SICK, werd er gekozen om zo veel mogelijk componenten van deze producent te gebruiken.

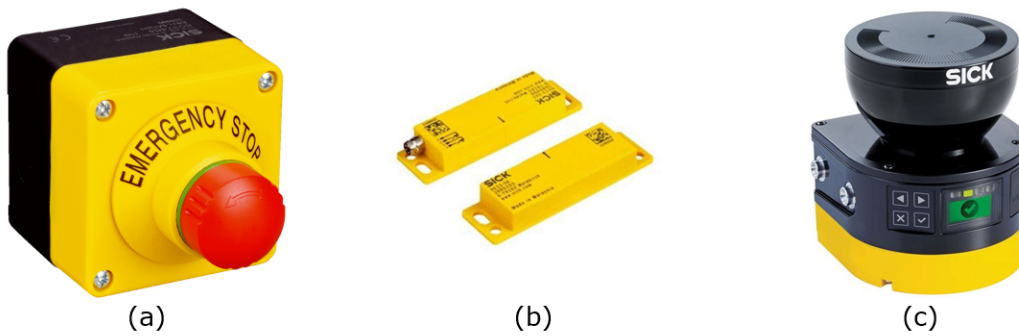
Als centrale veiligheidscontroller is er gekozen voor de Beckhoff EK1960 (figuur 5.3). Verder is op deze controller ook EtherCAT compatibiliteit voorzien zodanig dat de KELO wielen in een noodsituatie aangestuurd kunnen met een safe-torque-off signaal. Op deze centrale controller worden dan ook alle veiligheidsgerelateerde bedieningsorganen zoals noodstopknoppen en contactschakelaars aangesloten.

Voor dit project is de voorkeur gegeven aan contactloze, magnetische deurbewaking in de vorm van RE23-SA03 schakelaars (figuur 5.4 b). Deze bieden genoeg flexibiliteit en tolerantie bij de montage om enige speling geïntroduceerd door uitlijnfouten van de extrusieprofielen op te vangen.



Figuur 5.3: Beckhoff EK1960 veiligheidscontroller [27]

Voor veiligheidsscanners is er gekozen voor de MicroScan3 (figuur 5.4 c), deze biedt de meest optimale combinatie van functies voor dit project. Deze is voldoende compact (112 mm x 163 mm x 111,1 mm) om in de hoeken van het mobiele platform in te bouwen voor een optimale bescherming tegen accidentele aanrijdingen. Verder biedt deze naast zijn standaard veiligheidsuitgangen ook de mogelijkheid om ruwe meetdata te publiceren, wat navigatie en lokalisatie in een later stadium van het project aanzienlijk vereenvoudigd. De microScan3 is echter ook een relatief eenvoudig model en bezit niet de extra robuuste behuizing zoals de outdoorScan3 of de 3D LiDAR technologie van de Multiscan100-S.



Figuur 5.4: Veiligheidscomponenten: (a): noodstopdrukknop, (b): magnetische deurcontacten [41], (c): veiligheidslaserscanner [42]

Om veiligheidssignalen tussen het mobiele platform en het werkstation uit te wisselen, is het aangeraden om een tweede EK1100-terminal uitgerust met veilige I/O's te plaatsen in het werkstation. Op deze manier kunnen zowel de standaard signalen als de veiligheidssignalen over hetzelfde communicatiemedium gestuurd worden.

5.4.2 Besturing

De besturing van het gehele mobiele platform wordt gerealiseerd door zelf samengestelde computer van HP. Deze gebruikt de Ubuntu distro van Linux gebruiken als besturingssysteem. De keuze voor Ubuntu is gedreven door het gebruik van het ROS2-framework om de dataverwerking in de mobiele manipulator te optimaliseren.

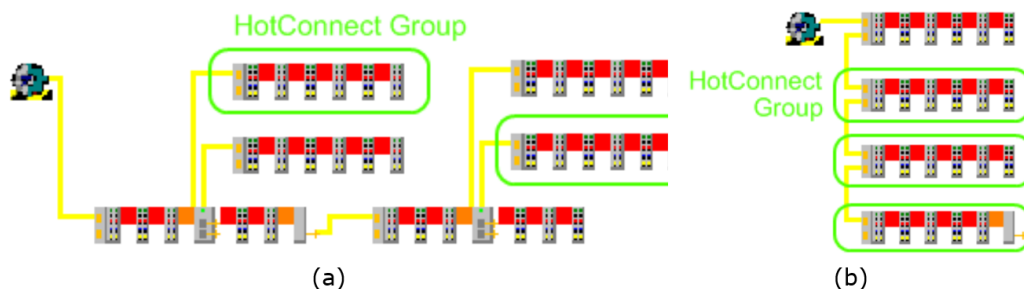
Verder dienen de componenten van de besturing die instaan voor de veiligheid van de machine ook aan enkele eisen te voldoen:

- moet bestand zijn tegen de normale bedrijfsbelasting en tegen invloeden van buitenaf,
- een storing in de apparatuur of de programmatuur van het besturingssysteem mag niet tot een gevaarlijke situatie leiden,
- fouten in de besturingslogica mogen niet tot een gevaarlijke situatie leiden,
- redelijkerwijs voorzienbare menselijke fouten gedurende de werking mogen niet tot een gevaarlijke situatie leiden.

[25, p. 27]

5.4.3 Communicatie

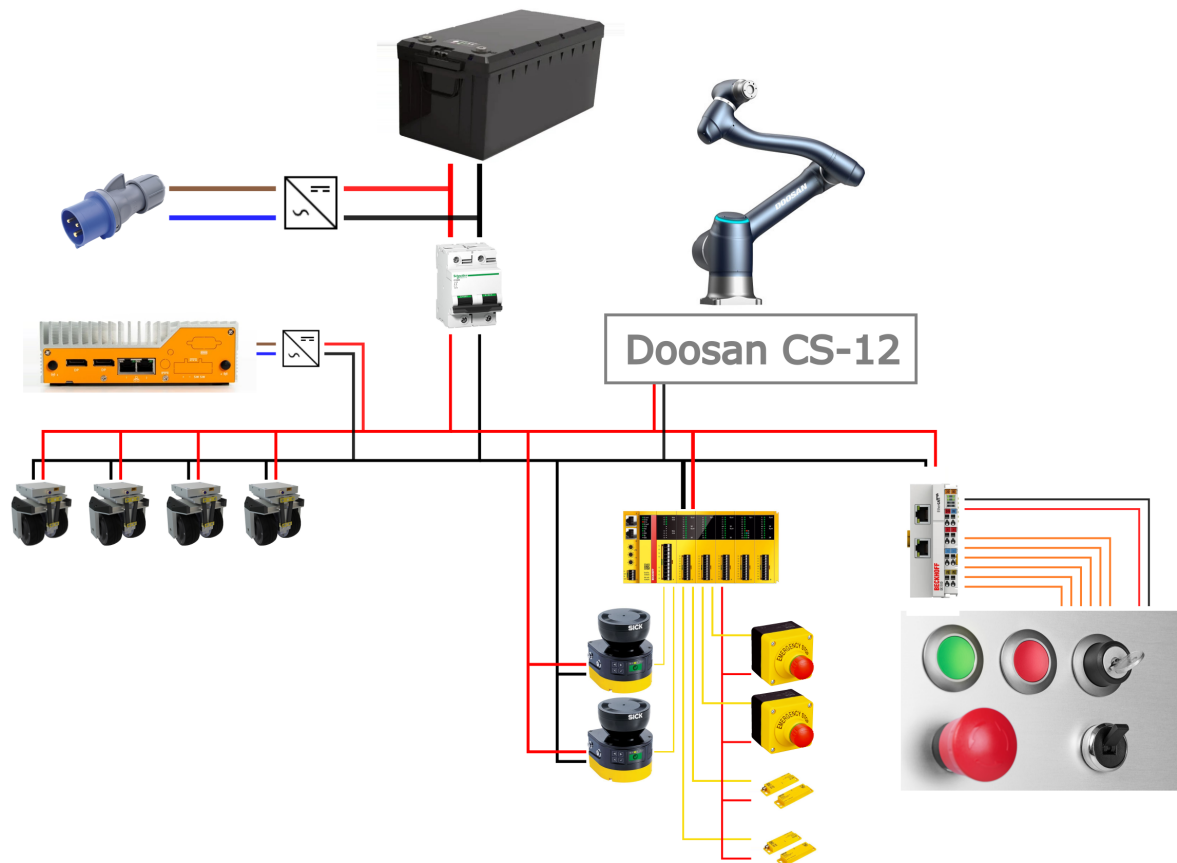
Aan een workstation moeten het mobiele platform en de cobot ook kunnen communiceren met eventuele machines die bij het station horen. Ook dit is realiseerbaar met het EtherCAT-netwerk door middel van Hot Connect. Hot Connect binnen een EtherCAT-netwerk laat vooraf geconfigureerde toestellen vrij aan of af te koppelen aan het netwerk terwijl dit operationeel is. Het is toepasbaar in combinatie met verschillende netwerkenconfiguraties zoals de traditionele ster (figuur 5.5 a) of lijn (figuur 5.5 b)) topologie. Het is echter niet mogelijk redundante verbindingen te leggen met de EtherCAT-slaves die in een Hot Connect groep zitten.



Figuur 5.5: EtherCAT Hot Connect netwerktopologieën [43]: (a) sterconfiguratie, (b) lijnconfiguratie

5.5 Elektrische schema's

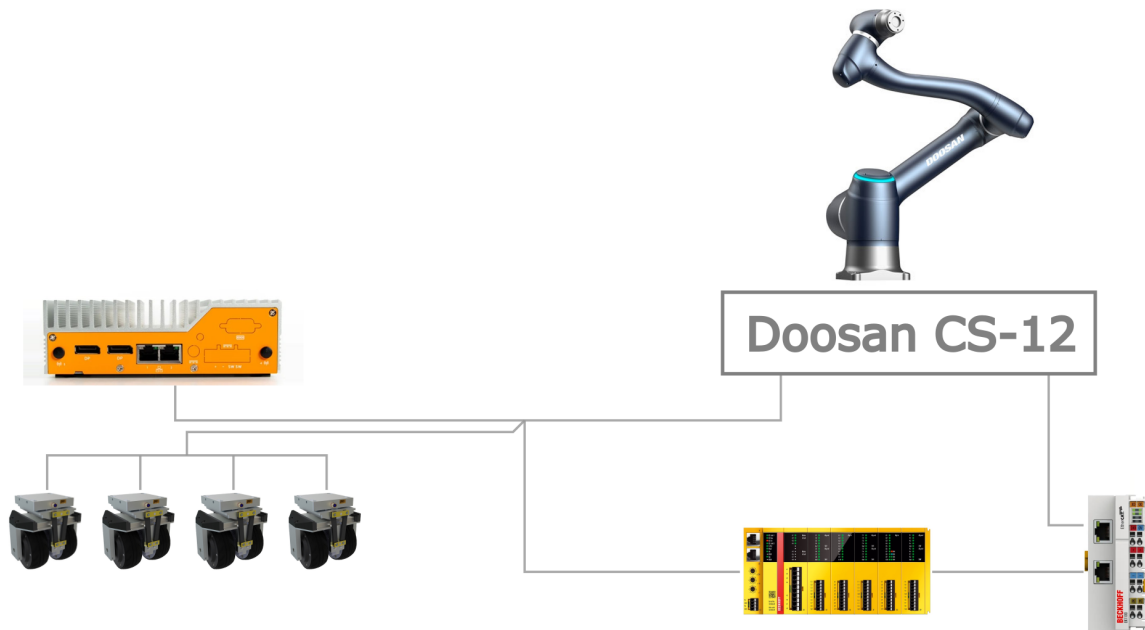
Als alle componenten voor hun doeleinden gekozen zijn, dienen deze samengebouwd te worden op het mobiele platform. Om een algemeen overzicht te krijgen van alle componenten die aangesloten dienen te worden op de batterij, werd er een principeschema (figuur 5.6) opgesteld. Dit laat toe de fysieke installatie van de componenten te versnellen om zo sneller in de testfase te kunnen geraken. Op basis van deze schema's kunnen dan achteraf de correcte elektrische schema's opgesteld worden om de realiteit adequater te weerspiegelen.



Figuur 5.6: Principeschema elektrische componenten

De centrale component in dit schema is de batterij. Deze voedt alle elektrische componenten die op het mobiele platform aanwezig zijn. Als eerste component na de batterij is er een hoofdschakelaar voorzien om de gehele installatie spanningsloos te stellen alvorens er werken aan uitgevoerd worden. Verder zijn ook de vier KELO drive 100 modules terug te vinden, de cobotcontroller die op zijn beurt de cobotarm voedt, de veiligheidscontroller met bijbehorende sensoren alsook de EtherCAT-terminal waarop nog enkele bediendingsorganen en controlelampen zijn aangesloten. Er is ook een gelijkrichter aanwezig die de batterijlader voorstelt. Deze zal de wisselstroom van het net omvormen naar gelijkstroom om op te slaan in de batterij. Daarnaast is er ook een wisselrichter voorzien om de centrale controller te voeden aangezien deze wisselspanning nodig heeft en dient de functioneren op de momenten dat het platform niet aan een werkstation gekoppeld is. OnLogic biedt ook de optie aan de computer met gelijkspanning te voeden en dan is de extra wisselrichter overbodig.

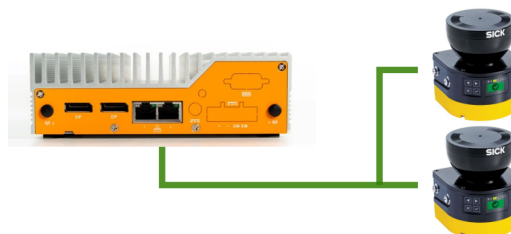
Ook voor de andere bedrading zoals alle componenten die op de EtherCAT-bus (figuur 5.7) zijn aangesloten, zijn soortgelijke principeschema's opgesteld. Voor het Ethernetnetwerk, weergegeven in figuur 5.8, is dit beperkt daar dit enkel gebruikt wordt om de actuele meetdata van de LiDAR-veiligheidsscanners door te sturen.



Figuur 5.7: Principeschema EtherCAT-bus

Het EtherCAT-schema is een stuk beperkter dan het elektrische schema maar bevat veelal dezelfde componenten. Het gehele netwerk is opgebouwd uit de centrale controller, de veiligheidscontroller, de cobotcontroller, de EtherCAT-terminal en de vier KELO drive 100 modules.

Eventuele I/O's van het werkstation kunnen met een externe EtherCAT-terminal aangesloten worden op het netwerk van de mobiele manipulator.



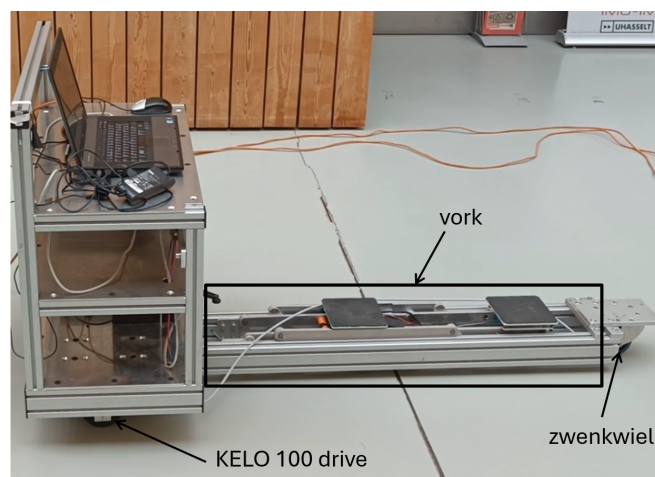
Figuur 5.8: Principeschema Ethernetnetwerk

De enige componenten die gebruik maken van het ethernet netwerk zijn de centrale controller en de veiligheidsscanners om ruwe meetgegevens uit te wisselen.

Hoofdstuk 6

Aansturing mobiel platform met twee KELO-drives

Zodra alle elektronica gemonteerd is, kan worden gestart met de ontwikkeling van software om het mobiel platform aan te sturen. Dit gebeurt door middel van een computer met het Ubuntu-besturingssysteem, om optimaal gebruik te maken van het ROS2-framework. Hiermee worden de KELO-drives op het platform aangestuurd. Voor de KELO-drives is er op de GitHub-pagina van KELO-robot al bestaande open source ROS1-code om deze aan te sturen via EtherCAT. Een nadeel van deze code is dat deze ontwikkeld werd voor ROS1. Dit is niet het gewenste formaat wat een omzetting naar ROS2 noodzakelijk maakt. ACRO heeft in het verleden al een gelijkaardig platform ontwikkeld dat beschikt over twee KELO-drives. Dit mobiel platform is origineel ontworpen om een ziekenhuiskar te verplaatsen [44]. Om verwarring te vermijden wordt dit platform verder ziekenhuisrobot genoemd. Voor het gebruik van dit platform is de code al omgezet naar ROS2. Naast de open-source software is er ook closed-source software, deze functioneert volgens koppelsturing in tegenstelling tot de open-source software die functioneert op basis van snelheidssturing. Figuur 6.1 geeft de opstelling van de ziekenhuisrobot weer.



Figuur 6.1: Opstelling ziekenhuisrobot [44]

De ziekenhuisrobot bevat achteraan de twee KELO-drives met daarboven een batterij en een EtherCAT-module voor communicatie met de wielmodules. Vooraan heeft deze een zwenkwiel, tussen deze en de wielmodules ligt dan het centerpunt. De ziekenhuisrobot heeft als functie om met de vork onder een ziekenhuiskar in te rijden en deze te vervoeren.

De ziekenhuisrobot biedt de mogelijkheid om al vroegtijdig met de code aan de slag te gaan en te achterhalen welke aanpassingen er nodig zijn om de code toe te passen op het mobiele platform.

6.1 Werking van de bestaande ROS-code

Een goede eerste stap is het begrijpen wat er in de bestaande ROS2-code gebeurt, dit zodat hier efficiënt mee samengewerkt kan worden en zodat voorkomen wordt dat functionaliteiten dubbel worden geïmplementeerd. De code door KELO-robile behandelt de verwerking van de snelheid. Deze wordt bij hoge waardes lineair verhoogd zodat er geen te grote versnelling naar de wielmodules gestuurd wordt. Daarnaast is deze ook begrenst binnen de ingestelde maxima en minima. Om de snelheid door te sturen naar de wielmodules wordt deze omgerekend naar setpoints. Ook de verbinding en communicatie via EtherCAT wordt hierin behandeld. De frequentie waarmee er gecommuniceerd wordt met de wielmodules is $20Hz$ oftewel een interval van vijftig milliseconden tussen stuursignalen.

Het project van de ziekenhuisrobot biedt toevoegingen aan de bestaande code voor communicatie en aansturing van de KELO-drives. De belangrijkste toevoeging is een hoofdsript genaamd "PlatformDriver". Deze voegt onderstaande functionaliteiten toe:

- **Parameters:** Staat toe om algemene parameters zoals maximale snelheden en versnellingen in te stellen;
- **Odometry:** Gebruikt de waardes afkomstig uit de encoder en standaard eigenschappen, zoals de afmetingen, om de odometrie te berekenen voor het platform. Deze worden dan gepubliceerd en omvatten de snelheid in de X- en Y-richting alsook de hoeksnelheid;
- **IMU:** Publiceert de IMU-waarden afkomstig uit de twee wielmodules, waaronder accelerometer- en gyroscoopwaarden;
- **CmdVel:** De subscriber die luistert naar uitgestuurde twistcommando's om de gewenste snelheden door te geven aan de rest van het script;
- **Joy:** De subscriber die luistert naar de joystick-input voor besturing van het platform;
- **Overige publishers:** Deze geven overige informatie zoals fout meldingen of status van de batterij.

Deze gepubliceerde gegevens kunnen vervolgens worden weergegeven in de terminal of door andere scripts gebruikt worden om verder te verwerken. Om een duidelijk overzicht te bewaren, is er voor gekozen om een nieuw script aan te maken die de parameters inleest, verwerkt en de corresponderende stuursignalen via twistcommando's publiceert. De parameters en stuursignalen worden verduidelijkt in de loop van het hoofdstuk.

Naast het hoofdsript zijn er ook enkele kleinere scripts toegevoegd voor de integratie met de EtherCAT-module, specifiek de EK1100 en EK1122 van Beckhoff. Dit is echter informatief en wordt verder niet gebruikt.

6.2 Analyse

Nadat de werking van de code gekend is, kan worden gestart aan een eerste analyse. Deze heeft als doel om te kijken welke waarden de sensoren in elk van de wielmodules aanleveren. Tabel 6.1 omvat de verschillende parameters die uitgelezen worden en verder in de analyse en berekeningen gebruikt zijn.

Tabel 6.1: Parameters uit interne sensoren KELO drive 100

Parameter	Omschrijving	Eenheid
sensor_ts	Tijdstip waarop sensorwaarden zijn opgevraagd	ns
encoder	Stand van de encoder van de motor	rad
velocity	Snelheid van de encoder van de motor	rad/s
current_q	Kwadratuurstroom van de motor	A
encoder_pivot	Stand van de encoder van het zwenkmechanisme	rad
velocity_pivot	Snelheid van de encoder van het zwenkmechanisme	rad/s
accel_x	IMU accelerometer volgens x-as	m/s ²
accel_y	IMU accelerometer volgens y-as	m/s ²
accel_z	IMU accelerometer volgens z-as	m/s ²
gyro_x	IMU gyroscoopwaarden volgens x-as	rad/s
gyro_y	IMU gyroscoopwaarden volgens y-as	rad/s
gyro_z	IMU gyroscoopwaarden volgens z-as	rad/s

Uiteraard zijn niet alle parameters even bruikbaar of worden deze allemaal gebruikt bij het finaal aansturen van het platform. Bovenstaande parameters lijken bij een eerste hypothese nuttig om te gebruiken maar na verdere analyse zijn sommige overbodig en is het toch handig om te kijken waarom.

Onderstaande delen bespreken ten eerste de analyse en de conclusies die bekomen zijn op basis van observaties van de waardes onder normale operatie. Hierbij is ook aangehaald welke parameters uiteindelijk overbodig zijn. Ten tweede wordt een analyse gedaan waarbij er tijdens normale operatie tegen het ziekenhuisrobot een externe kracht uitgeoefend wordt, zowel in mee- als tegenwerkende richting. Zo worden de verschillen en gelijkenissen ten opzichte van normale operatie in kaart gebracht en gebruikt om te detecteren of er een externe kracht op de kar wordt uitgevoerd en in welke richting.

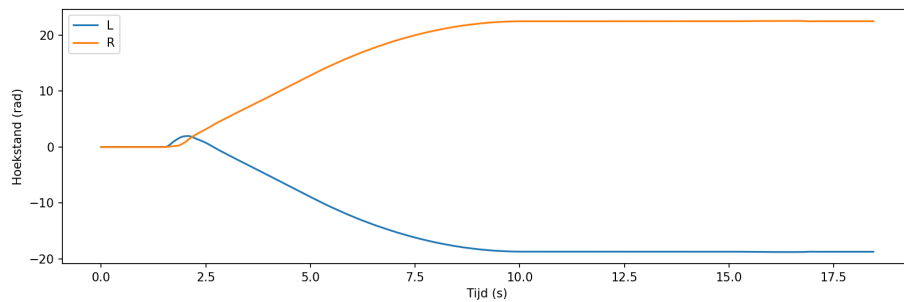
6.2.1 Normale operatie

Voor de analyse onder normale operatie werden er vaste snelheidssignalen naar de wielmodules gestuurd om een zo goed mogelijke vergelijking te maken met verdere metingen. De wielmodules werden als volgt aangestuurd:

- **Gedurende de eerste 1,5 seconden:** snelheid $vx = 0m/s$, dit ter controle en om te evalueren wat de waarde van een parameter is bij start;
- **Volgende 3,5 seconden:** constante snelheid van $vx = 0,2m/s$;
- **Volgende 5 seconden:** geleidelijk vertragen van $vx = 0,2m/s$ naar $vx = 0m/s$.

Omdat een enkele meting 28 grafieken produceert, dit vanwege de twee modules met elk twee wielen, worden enkel de belangrijkste besproken. De grafieken die niet besproken worden betreffen deze van de accelerometers en gyroscopen. Deze hebben echter ruis wat het moeilijker maakt om conclusies uit te trekken. Daarnaast wordt er enkel voor één module de waarden van een parameter besproken doordat deze voor beide modules gelijkaardig zijn.

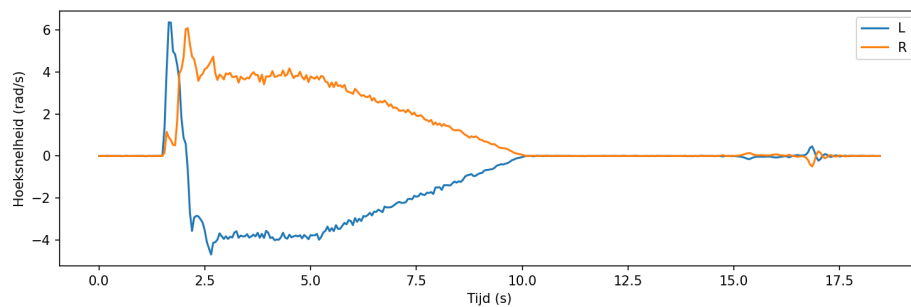
De eerste parameter *sensor_ts* komt op twee manieren te pas, enerzijds om telkens als x-as op de grafieken te gebruiken zodat alles gelijk weergegeven wordt en anderzijds om te controleren hoe accuraat de frequentie van $20Hz$ gevolgd wordt. Voor de tweede parameter encoder is onderstaande grafiek in figuur 6.2 bekomen.



Figuur 6.2: Stand encoders hoek(rad) i.f.v. tijd(s) voor module B

Een eerste vaststelling is dat de twee wielen van één module een tegengesteld resultaat geven, dit vanwege de manier waarop deze gemonteerd staan op de module. Ook is er voor gekozen deze waarde niet te inverteren voor het weergeven van de grafieken. Daarnaast is de hoek niet modulair met 2π , waardoor deze waarden hoger oplopen dan 2π .

De grafiek van de volgende parameter, velocity (hoeksnelheid), is weergegeven in figuur 6.3.



Figuur 6.3: Hoeksnelheid(rad/s) i.f.v. tijd(s) voor module B

Bij start nemen het linker en rechter wiel een gelijke hoeksnelheid aan. Dit is te zien bij 2 seconden wanneer het startsignaal gegeven is. Dit betekent dat de wielmodule nog niet correct georiënteerd is om voorwaarts te bewegen. Deze hoeksnelheid zorgt er voor dat de module rond zijn centrale zwenkmechanisme roteert tot dat de gewenste oriëntatie bereikt is.

Hier is het stuursignaal duidelijk waarneembaar met tussen 5 seconden en 10 seconden een gelijkmatige daling naar 0. Tussen 15 seconden en 17,5 seconden is er ook een kleine schommeling te zien. Hier is er op het platform een externe kracht uitgeoefend om een eerste reactie te zien van deze parameter.

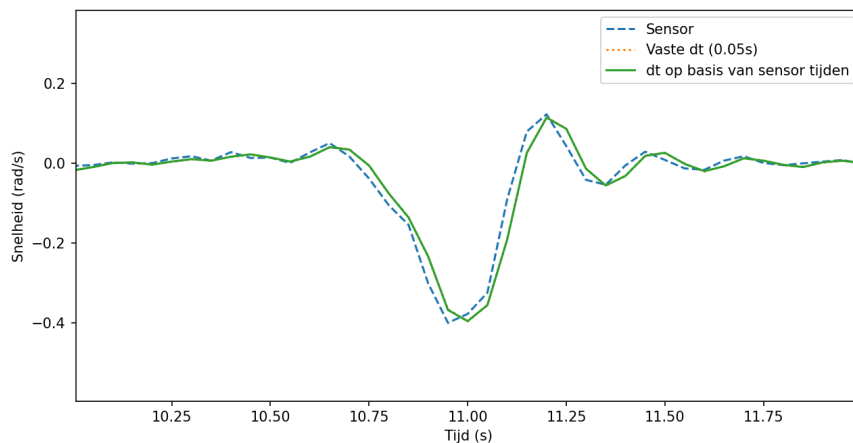
Het is ook mogelijk om de hoeksnelheid te berekenen aan de hand van de vaste frequentie van $20Hz$ of de parameter *sensor_ts* samen met de waarde van de encoder uit figuur 6.2. De hoeksnelheid volgens de frequentie van $20Hz$ is volgens onderstaande formule berekend:

$$\omega(t_i) = \frac{d\theta_i}{dt} = \frac{\theta(t_i) - \theta(t_{i-1})}{0,05s} \quad (6.1)$$

Voor het berekenen van de hoeksnelheid op basis van de sensor tijdstippen dient de noemer aangepast te worden volgens onderstaande formule:

$$\omega(t_i) = \frac{\theta(t_i) - \theta(t_{i-1})}{t_{sen,i} - t_{sen,i-1}} \quad (6.2)$$

De drie bekomen snelheden worden als volgt op dezelfde grafiek geplott om zo verschillen op te merken en te kiezen welke snelheid het beste is om mee verder te rekenen. Deze zijn weergegeven in figuur 6.4.

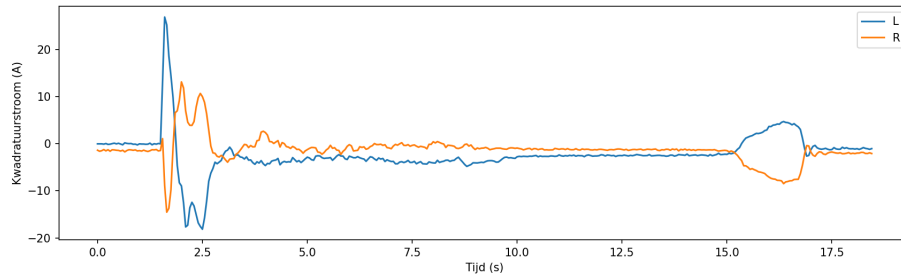


Figuur 6.4: Drie hoeksnelheid(rad/s) i.f.v. tijd(s) voor module B links

Bovenstaande grafiek toont een sectie van de volledige grafiek, de waardes van de drie hoeksnelheden verschillen echter minimaal van elkaar. De hoeksnelheden op basis van de vaste frequentie en sensortijdstippen overlappen volledig wat betekent dat in verdere berekening er met deze vaste frequentie gerekend wordt doordat dit de berekeningen vereenvoudigt en er geen grove fouten bekomen worden.

De beste hoeksnelheid om verder te gebruiken is deze afkomstig uit de sensoren in de wielen. Deze loopt één tijdstip voor wat tot tijdigere reacties leidt en vereist geen extra berekeningen. Dit heeft ook tot gevolg dat de parameter encoder verder geen toepassingen heeft. In verdere berekeningen voor de odometrie en schatting van het koppel kan er rechtstreeks met de parameter *velocity* gewerkt worden.

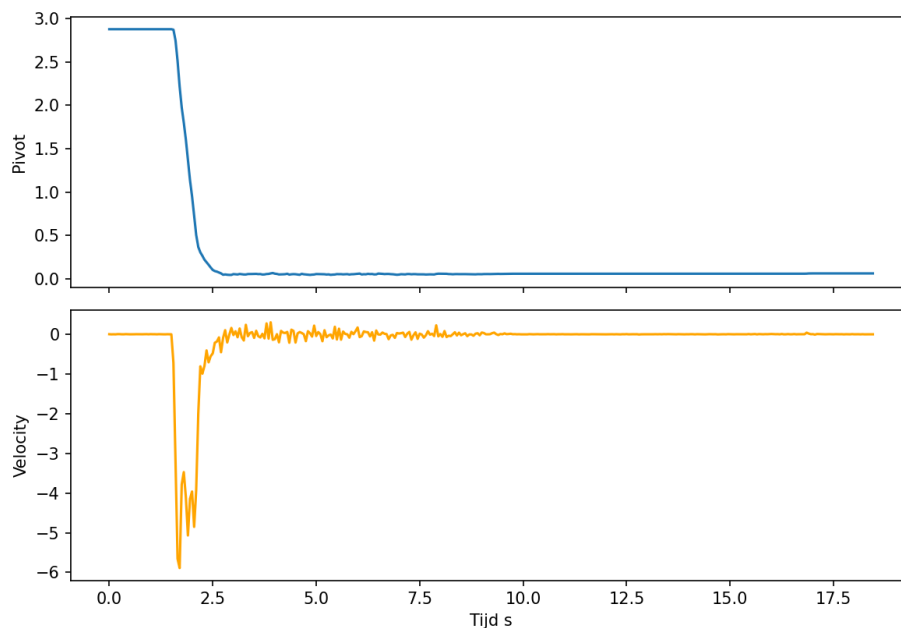
Daarnaast is er de parameter van de *kwadratuurstroom*. Deze is vrijwel de belangrijkste parameter aangezien deze een evenredig verband heeft met het koppel. Dit volgens $i_q \cdot k_t \approx \tau$ met k_t de koppel-constante bepaald uit de technische specificaties van de KELO 100 drives. Onderstaande figuur 6.5 geeft de grafiek van de *kwadratuurstroom*.



Figuur 6.5: Kwadratuurstroom(A) i.f.v. tijd(s) voor module B

Deze grafiek bevat echter nog enkele onzuiverheden. Als eerst zijn er de grote pieken rond 2,5 seconden. Komende figuur 6.6 geeft hiervoor de verduidelijking met de stand en de hoeksnelheid van de centrale encoder waar de module rond roteert. Naast deze pieken is er inductance windup, bij bovenstaande grafiek is dit minimaal waarneembaar maar bij komende grafieken van andere metingen wordt dit al opvallender. Hierbij treed er bij stilstand nadat de modules aangestuurd zijn geweest een opbouw van inductantie op waardoor de kwadratuurstroom nog hoog blijft en slechts langzaam daalt. Als laatste hebben de wielen een houdkoppel, dit is tussen 15 en 17,5 seconden zichtbaar wanneer er tegen het platform geduwd wordt en er een significante stijging van de stroom gemeten werd om deze kracht tegen te werken en er voor te zorgen dat het platform op zijn huidige locatie blijft staan. De verwerking van deze onzuiverheden wordt na de analyse besproken.

Figuur 6.6 geeft de grafieken voor de hoekstand en hoeksnelheid van de centrale encoder waar de module rond roteert. De hoek stand is hier, in tegenstelling tot de encoders van de wielen, wel modulair met 2π .



Figuur 6.6: Hoekstand pivot(rad) en hoeksnelheid pivot(rad/s) i.f.v. tijd(s) voor module B

Deze grafiek assisteert bij het verklaren van de pieken die telkens optreden bij zowel de hoeksnelheid als de kwadratuurstroom. Omdat telkens bij het uitvoeren van de metingen het platform terug handmatig verplaatst werd naar zijn startpositie, waren de wielmodules niet langer naar voren gericht. Bij het sturen van een snelheid dienen deze terug de correcte oriëntatie aan te nemen, dit aan de hand van het snelheidsverschil tussen het linker en rechter wiel. Op figuur 6.3 is dit zichtbaar tussen 1,5 en 2,5 seconden waar het linkerwiel eerst een tegengestelde hoeksnelheid (door spiegeling is dit op de grafiek zichtbaar als dezelfde hoeksnelheid) heeft als het rechter wiel. Ook de aanpak en verwerking hiervan wordt na het analyseren van het effect van externe aangelegde krachten besproken.

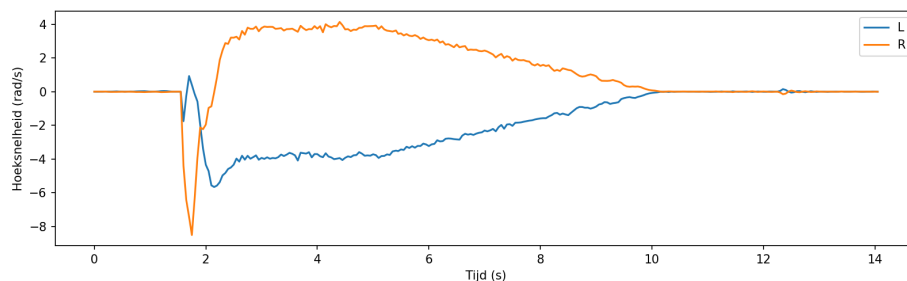
6.2.2 Aanleggen van externe krachten

Nu duidelijk is hoe de grafieken van de verschillende parameters er dienen uit te zien bij normale operatie, is het eenvoudig om de verschillen of gelijkenissen tussen verschillende metingen op te merken. Aan de hand hiervan kan dan de code ontwikkeld worden voor het detecteren van externe krachten en het platform hierop te laten reageren. Onderstaande metingen bespreken de grafieken wanneer er een meewerkende kracht, dus volgens de richting waarin het platform voortbeweegt, en een tegenwerkende kracht aangelegd worden.

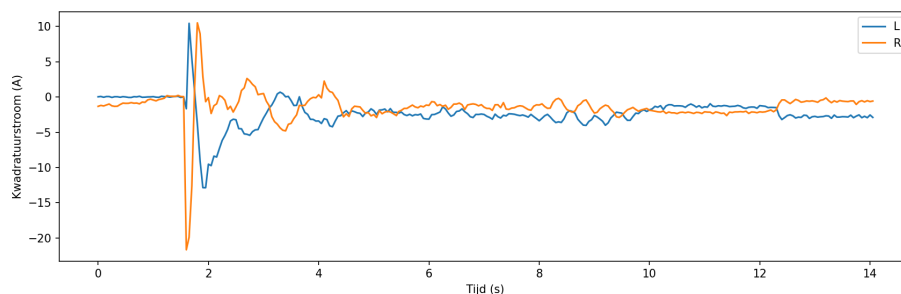
De grafieken die volgen uit de parameters *encoder*, *encoder_pivot* en *velocity_pivot* hebben geen opmerkelijke verschillen met deze bij normale operatie.

Meewerkende kracht

De tweede meting werd uitgevoerd met een meewerkende kracht. Hierbij werden voor de parameters *velocity* en *current_q* respectievelijk de grafieken van figuur 6.7 en figuur 6.8 bekomen.



Figuur 6.7: Hoeksnelheid(rad/s) i.f.v. tijd(s) voor module B



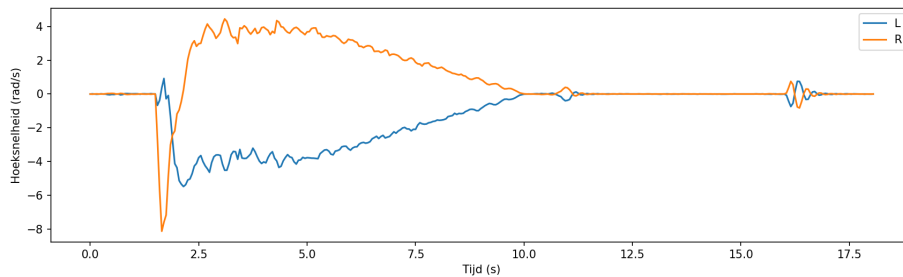
Figuur 6.8: Kwadratuurstroom(A) i.f.v. tijd(s) voor module B

Ook bij deze grafieken zijn er pieken tussen 1,5 en 2,5 seconden waarneembaar, deze zien er anders uit dan deze bij de normale operatie. Dit verschil treedt op doordat de oriëntatie die de wielmodules hebben bij start anders zijn dan deze van de meting bij normale operatie.

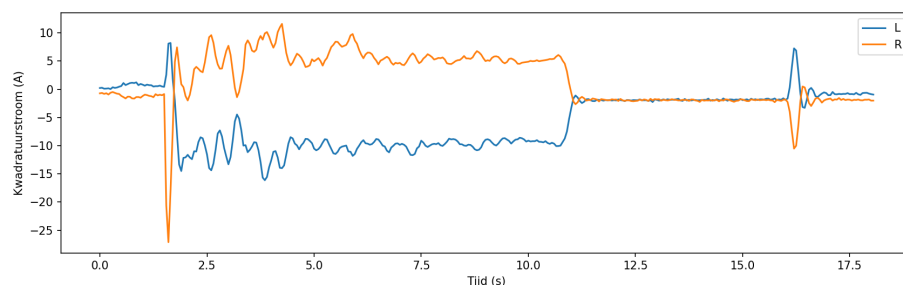
Verder is er op de grafiek van de kwadratuurstroom zichtbaar dat de waarden van het linker- en rechterwiel dichter bij elkaar liggen. Dit komt overeen met een meewerkende kracht doordat er minder stroom dient geleverd te worden aan de wielen om de gewenste snelheid te behalen.

Tegenwerkende kracht

De derde meting werd uitgevoerd met een tegenwerkende kracht. Hierbij werden voor de parameters *velocity* en *current_q* respectievelijk de grafieken van figuur 6.9 en figuur 6.10 bekomen.



Figuur 6.9: Hoeksnelheid(rad/s) i.f.v. tijd(s) voor module B



Figuur 6.10: Kwadratuurstroom(A) i.f.v. tijd(s) voor module B

Net zoals de grafieken bij een meewerkende kracht zijn hier weer gelijkaardige pieken waar te nemen tussen 1,5 en 2,5 seconden. In tegenstelling tot voorgaande meting liggen nu de waarden van het linker- en rechterwiel verder uit elkaar voor de kwadratuurstroom.

Conclusie

Opvallend is dat de modules, zowel bij een meewerkende als een tegenwerkende kracht, hun gewenste snelheid blijven behouden. Deze eigenschap maakt het mogelijk om de parameter *velocity* te gebruiken om het benodigde koppel bij een bepaalde snelheid in te schatten. Door dit te vergelijken met het effectief gemeten koppel, kan er een verschil vastgesteld worden dat mogelijk informatie geeft over de aard van de uitgeoefende kracht. Op basis van dit verschil wordt in het volgende deel verder onderzocht of er een resulterende koppel aanwezig is: verwacht wordt dat deze bij normale werking nul benadert, terwijl een meewerkende kracht zou kunnen leiden tot een negatieve afwijking, en een tegenwerkende kracht tot een positieve.

6.3 Aanpassingen en toevoegingen

Om naar een uiteindelijk bruikbaar resultaat toe te werken dat vanuit de ruwe data vertrekt en omvormt, dient een stappenplan opgesteld te worden. Dit omvat de nodige formules, berekeningen en aannames. Doordat de omvorming via de samenhang tussen de verschillende parameters werkt is het niet mogelijk om dit individueel per parameter te bespreken maar eerder in de volgorde waarop dit in de code wordt uitgevoerd.

Het finaal resultaat wordt vervolgens gebruikt voor de ziekenhuisrobot om de correcte acties te ondernemen. De lijst met mogelijkheden waarop het platform kan reageren op bepaalde waarden van het finaal resultaat zijn bij het deel van de acties ook aangehaald. Uit deze acties is ook een gebruiksaanwijzing opgesteld zodat het eenvoudiger is om met het platform te werken.

Finaal wordt er als laatste toevoeging de menselijke factor in rekening gebracht. Het is niet mogelijk en een onrealistische verwachting dat de gebruiker continu de nodige kracht gaat uitoefenen op het platform. Daarom is er deze laatste stap nodig om gebruik van het platform te vergemakkelijken.

6.3.1 Methode

Verwerking

Een eerste stap voor het verkrijgen van een bruikbaar resultaat is het omzetten van de parameters naar koppel. Voor de kwadratuurstroom kan dit worden berekend met de koppelconstante van de motor k_t , die is bepaald door middel van de datasheet van de KELO 100 drive [14]. Onderstaande formule geeft de berekening voor deze constante:

$$k_t = \frac{\tau}{I} = \frac{4,8Nm}{20A} = 0,24 \frac{Nm}{A} \quad (6.3)$$

Als volgende stap wordt het koppel voorspeld aan de hand van de hoeksnelheid, dit om normale operatie duidelijk te onderscheiden van uitgeoefende krachten. Om te zien hoe de koppel-snelheidsrelatie eruitziet wordt gekeken naar de motorspecificatie van de KELO 100 drive. Deze gebruikt DC-motoren dus wordt er gewerkt via de corresponderende koppel-snelheidscurve. Hiervoor dienen enkele waarden gekend te zijn: T_{stall} is het koppel bij stilstand waarbij de motor maximaal belast is, en ω_{free} is de maximale hoeksnelheid zonder last. In de testfase zijn deze waarden experimenteel vastgesteld met $T_{stall} = 0,5Nm$ en $\omega_{free} = 14rad/s$. Het bepalen van het voorspelde koppel wordt gerealiseerd met onderstaande formule:

$$\tau_{exp} = T_{stall} \cdot \left(1 - \frac{|\omega|}{\omega_{free}}\right) \cdot sign(\omega) \quad (6.4)$$

Deze formule is echter onstabiel bij lage snelheden door lichte ruis. De oorzaak hiervoor is dat de snelheid niet perfect nul kan zijn. Deze waarde schommelt sterk rond het nulpunt waardoor deze soms snel wisselt tussen negatief en positief. Als gevolg wordt er voor gekozen de methode die gebruik maakt van het verwacht koppel (τ_{exp}) enkel te implementeren zodra er een eenduidige snelheid gemeten wordt.

Hieruit volgt ook de volgende stap om de inductantieopbouw weg te werken. Deze wordt weggevoerd door met de voorgaande waarde van de kwadratuurstroom het koppel te bepalen en het verschil te nemen met de huidige waarden. Dit gebeurt enkel indien er geen snelheid is en het verschil tussen de huidige en voorgaande kwadratuurstroom binnen het ingestelde bereik van $0,5A$ valt.

Als voorlaatste stap dienen de pieken veroorzaakt door de wielmodules, die zich bij start correct oriënteren, weggevoerd te worden. Er is voor gekozen om dit te doen door, indien de parameter *velocity_pivot* een hoeksnelheid groter als $0,5rad/s$ geeft. Het koppel te negeren en de waarde $0Nm$ te geven.

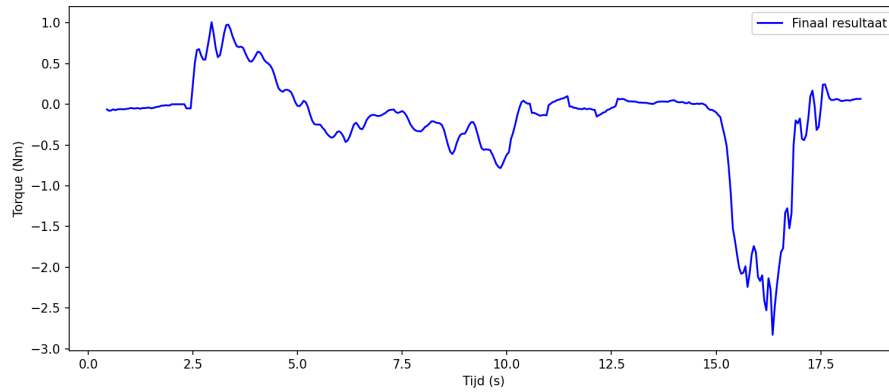
In de laatste stap worden alle berekende koppels voor de vier motoren samen opgeteld door onderstaande formule. De linker motoren dienen echter nog geïnverteerd te worden.

$$\tau_{tot} = \tau_{A_R} + (-\tau_{A_L}) + \tau_{B_R} + (-\tau_{B_L}) \quad (6.5)$$

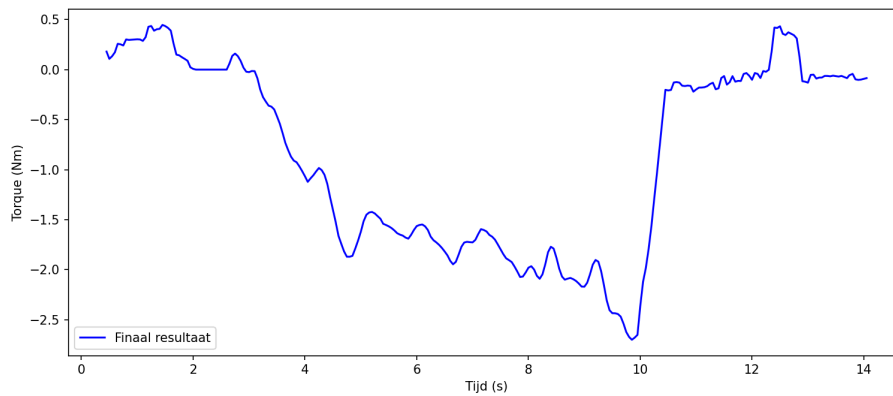
Op dit totale koppel wordt dan de detectie uitgevoerd om te bepalen of er een tegenwerkende of meewerkende kracht op het platform uitgevoerd wordt.

Bepaling detectiegrenzen

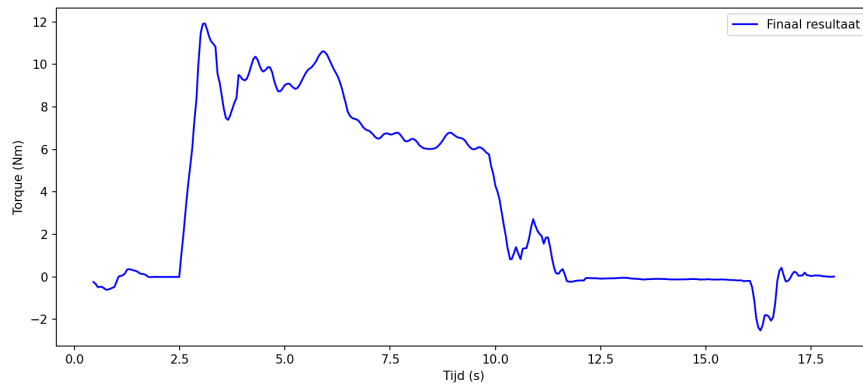
Het uitvoeren van de stappen om het totale koppel te berekenen geeft voor normale operatie, tegenwerkende en meewerkende kracht respectievelijk de grafieken in figuren 6.11 , 6.12 en 6.13.



Figuur 6.11: Finaal koppel normale operatie



Figuur 6.12: Finaal koppel tegenwerkende kracht

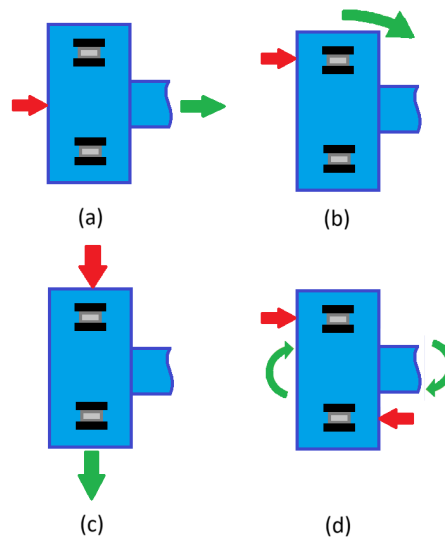


Figuur 6.13: Finaal koppel meewerkende kracht

Aan de hand van de bovenstaande grafieken wordt er gekozen om op $\tau = 2Nm$ de grens in te stellen voor een licht tegenwerkende kracht en op $\tau = -2Nm$ de grens voor een licht meewerkende kracht.

Acties

Vervolgens worden nu de acties bepaald die de ziekenhuisrobot uitvoert in bepaalde situaties. Figuur 6.14 geeft de verschillende situaties weer waar het platform zich in kan bevinden. De rode pijl is telkens de uitgeoefende kracht door de gebruiker, de groene pijl stelt de reactie van het platform op de corresponderende kracht voor.



Figuur 6.14: Situaties ziekenhuisrobot: (a) voorwaartse beweging; (b) draaien; (c) zijwaartse beweging; (d) rond centerpunt draaien

- **Voorwaartse beweging:** bij gelijke kracht ondervonden door beide wielmodules beweegt het platform voorwaarts of, indien tegengestelde kracht, achterwaarts.
- **Draaien:** bij een krachtenverschil tussen de wielmodules wordt er in de corresponderende richting gedraaid.

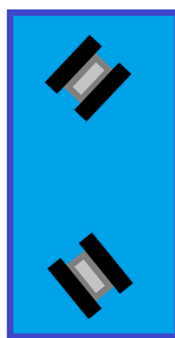
- **Zijwaartse beweging:** vergelijkbaar aan de voorwaartse beweging wordt er volgens de corresponderende kracht naar links of rechts bewogen. De kinematica van de wielmodules maakt het echter niet mogelijk om een kracht loodrecht op de wielen waar te nemen, de oplossing hiervoor wordt verder aangehaald.
- **Rond centerpunt draaien:** zodra een module een even grote, maar tegengestelde kracht ondervindt dan de andere module, draait het platform rond zijn centerpunt.

Voor bovenstaande situaties worden vervolgens de nodige meetbare parameters gezocht die het mogelijk maken deze situaties van elkaar te onderscheiden. Hoe deze parameters met elkaar samenwerken wordt finaal duidelijk gemaakt door overzichtelijke tabellen. Eerst zijn er nog twee obstakels die behandeld dienen te worden om de keuze van de parameters te vergemakkelijken.

Zoals al te zien was bij voorgaande grafieken, is het mogelijk dat de wielmodules bij start niet correct georiënteerd staan. Dit geeft als mogelijkheid dat een voorwaartse of achterwaartse kracht foutief geïnterpreteerd wordt indien de modules 180 graden of meer gedraaid staan. In deze situatie waar een voorwaartse kracht op het platform aangelegd wordt, gaat deze als achterwaartse kracht geïnterpreteerd worden. Het platform heeft daardoor geen relatief beeld over de oriëntatie van de modules. Hiervoor dient eerst een functie opgesteld te worden om de stand van de modules te detecteren.

De complicaties die hierbij te pas komen, is de encoder die modulair is met 2π . Op zichzelf is dit geen probleem maar de waarden van de wielmodules die overeenkomen met 0 graden, en dus vooruit, komen overeen met 3,195 rad en 4,796 rad. Dit leidt tot een complexere omvorming voor de detectie van het gebied tussen -90° en 90° , wat telt voor vooruit, en het gebied tussen 90° en 270° , wat telt voor achteruit. Deze functie wordt bij 3.3.2 over de implementatie in de code besproken.

Het tweede obstakel is, zoals aangehaald bij de zijwaartse beweging, dat door de kinematica van de wielmodules het onmogelijk is om een loodrechte kracht te meten. Als oplossing is hiervoor gekozen om bij stilstand de wielmodules onder een hoek van 45° te oriënteren, zoals in figuur 6.15.



Figuur 6.15: De oriëntatie van de wielmodules bij stilstand

Bij de oriëntatie voorgesteld in bovenstaande figuur is het nu mogelijk om zijwaartse krachten te detecteren. Een bijkomend obstakel is dat de resulterende koppel die de wielmodules meten voor voorwaartse en zijwaartse kracht identiek zijn. Ook geeft dit complicaties bij het detecteren om rond het centerpunt te roteren.

De gekozen oplossing hiervoor is om eerst voorwaarts van zijwaarts te onderscheiden. Hiervoor wordt de parameter *velocity_pivot* geïmplementeerd samen met het finale koppel, dit omdat de uitgeoefende kracht een moment veroorzaakt om de wielmodules waardoor de encoder van het zwenkmechanisme een hoeksnelheid waarneemt. De totale hoeksnelheid wordt door onderstaande formule bekomen:

$$\omega_{pivot_tot} = | \omega_{pivot_A} | + | \omega_{pivot_B} | \quad (6.6)$$

In de situatie waar een voldoende hoge hoeksnelheid waargenomen wordt, veranderen de wielmodules terug van oriëntatie naar deze van figuur 6.14. Indien het toch een zijwaartse kracht was die uitgeoefend werd, neemt het platform geen krachten waar en worden de modules terug naar de initiële oriëntatie gebracht. Indien er wel een kracht waargenomen wordt is dit als volgt een voorwaartse of roterende.

Om de ziekenhuisrobot te laten draaien is er ook een extra voorwaarde nodig. Het platform dient te draaien zodra er meer koppel aan module A of module B geleverd wordt. Daarvoor wordt een procentueel verschil bepaald tussen beide modules om te zien welke het meeste koppel ervaart ten opzichte van de andere. Hiervoor wordt eerst het totale absolute koppel per module berekend aan de hand van het eerder bekomen koppel per motor:

$$\tau_{A,abs} = | \tau_{A_R} | + | \tau_{A_L} | \quad (6.7)$$

$$\tau_{B,abs} = | \tau_{B_R} | + | \tau_{B_L} | \quad (6.8)$$

Als volgt wordt het percentage per module berekend, in deze situatie wordt dit alleen voor module B gedaan:

$$\tau_{\%B} = \frac{\tau_{B,abs}}{\tau_{B,abs} + \tau_{A,abs}} \cdot 100\% \quad (6.9)$$

Bovenstaande parameter is echter enkel van toepassing indien het platform al een voorwaartse beweging heeft. Om rond het centerpunt te roteren ondervinden de wielmodules een tegengesteld koppel die elkaar opheffen. Daarom wordt er in deze situatie niet met een percentage gewerkt maar met de absolute totaalwaarde van het koppel, dit volgens onderstaande formule:

$$\tau_{tot,abs} = \tau_{B,abs} + \tau_{A,abs} \quad (6.10)$$

Het absolute totaalkoppel dat bekomen wordt, wordt dan vergeleken met het totale koppel. Wanneer het totale koppel binnen een ingestelde grens rond nul ligt, maar het absolute totale koppel die grens overschrijdt, wijst dit erop dat er tegengestelde krachten worden uitgeoefend die elkaar opheffen. In dat geval wordt aangenomen dat het platform rond zijn eigen middelpunt roteert. Afhankelijk van de verdeling van de koppels kan dit linksom of rechtsom zijn.

De eerste overzichtstabel 6.2 geeft de voorwaarden voor de eerste stap van acties. De eerste drie kolommen zijn de parameters die gebruikt worden voor het onderscheiden van zijwaartse beweging en het overschakelen naar de tweede stap aan acties, waarvoor de wielmodules voorwaarts georiënteerd worden. De laatste drie kolommen zijn de waardes voor de aansturing van het platform.

Tabel 6.2: Eerste stap acties ziekenhuisrobot

Waargenomen totale torque (Nm)	Stand module (°)	ω_{pivot_tot} (rad/s)	$Vx(m/s)$	$Vy(m/s)$	$Va(rad/s)$
> 2	45	0	0	0,3	0
< -2	45	0	0	-0,3	0
> 1	45	$> 0,2$	Tweede	stap	acties
< -1	45	$> 0,2$	Tweede	stap	acties

In tabel 6.3 komen de tweede stap aan acties voor die volgen uit de twee laatste rijen uit bovenstaande tabel. Ook staat hier weer de reactie van de ziekenhuisrobot op de bijhorende situatie. De eerste vier kolommen zijn de parameters die opgemeten worden of berekend om de tweede stap aan acties van elkaar te onderscheiden. De laatste twee kolommen zijn de waardes die vervolgens naar het platform gestuurd worden.

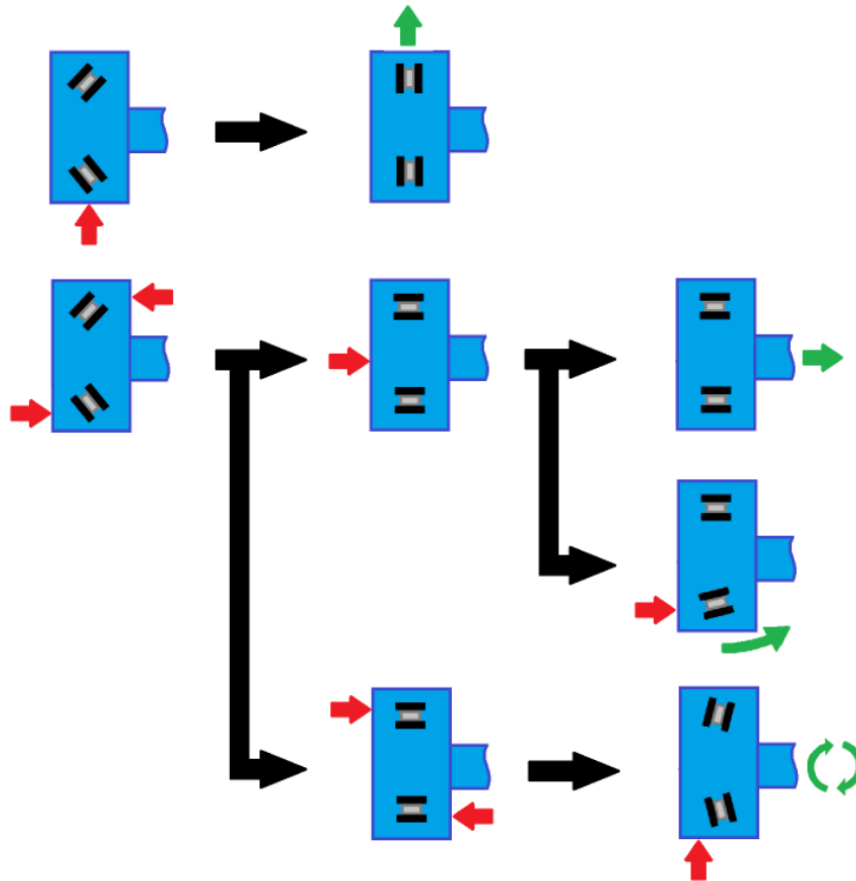
Tabel 6.3: Tweede stap acties ziekenhuisrobot

Waargenomen totale koppel (Nm)	Stand module (°)	Module B (%)	$\tau_{tot,abs}$ (Nm)	$Vx(m/s)$	$Vy(m/s)$	$Va(rad/s)$
> 2	0	50	< 6	-0,3	0	0
< -2	0	50	< 6	0,3	0	0
$< -1,5$	0	> 70	< 6	-0,15	0	0,3
$< -1,5$	0	< 30	< 6	-0,15	0	-0,3
tussen 0 en 1	0	/	> 6	0	0	0,3
tussen -1 en 0	0	/	> 6	0	0	-0,3
> 2	45	50	< 6	0	0	0,3
< -2	45	50	< 6	0	0	-0,3

Bovenstaande tabel geeft de overige acties die het platform uitvoert volgens de correcte voorwaarden van de parameters weer. De eerste twee rijen zijn respectievelijk het achterwaarts en voorwaarts rijden. Rij drie en vier worden enkel actief zodra het platform al een voorwaartse snelheid heeft en zijn respectievelijk voor het links en rechts draaien. Rij vijf en zes zijn respectievelijk voor het linksom- en rechtsom draaien rond het centerpunt. De laatste twee rijen zijn specialer, doordat rotatie de wielmodules van oriëntatie doet veranderen is het niet meer mogelijk om volgens rij vijf en zes te werken. Deze rijen worden enkel actief tijdens het roteren rond het centerpunt.

De waarden in bovenstaande tabellen zijn hoofdzakelijk experimenteel bepaald. Dit gebeurde aan de hand van testen met het platform, waarbij telkens gezocht werd naar een comfortabele, intuïtieve reactie van het platform op de acties. Parameters zoals de grenswaarden voor het koppel of hoeksnelheid werden afgeleid uit observaties en grafieken tijdens die testen. Ook de snelheden die naar het platform gestuurd worden, zijn bewust beperkt gehouden om een veilige en stabiele werking te garanderen.

Na het onderverdelen van de acties onder verschillende stappen is het makkelijker voor de gebruiker om een symbolische gebruiksaanwijzing op te stellen. Deze volgt uit figuur 6.14 en bekommt het schema van figuur 6.16.



Figuur 6.16: Gebruiksaanwijzing ziekenhuisrobot iteratie 1

Aanpak menselijke factor

Finaal wordt de menselijke factor bekeken en behandeld. Het platform wordt niet door machines bestuurd die continu een kracht resulterend in een koppel van 2 Nm kunnen aanleggen. Daarom wordt ook hier met een stappenplan gewerkt om afwijkingen en imperfecties toe te laten. Eerst worden de acties gegroepeerd:

- **x-richting:** deze groep omvat rij één en twee uit tabel 6.3;
- **y-richting:** deze groep omvat rij één en twee uit tabel 6.2;
- **rotatie:** deze groep omvat rij vijf en zes uit tabel 6.3.

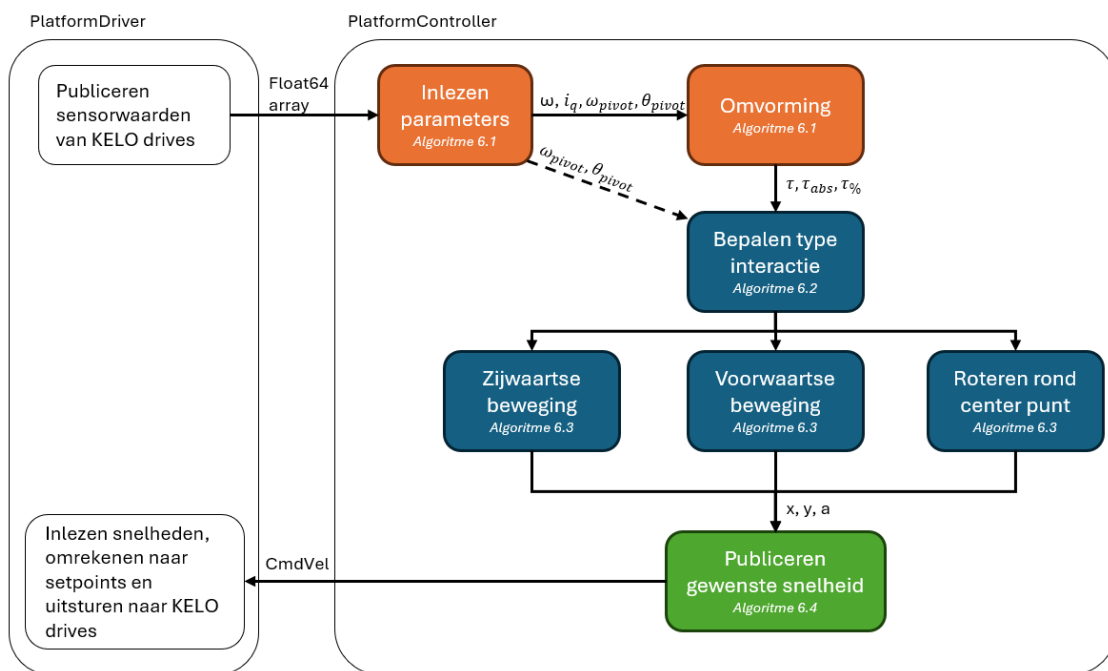
Zodra aan de voorwaarde binnen een groep voldaan wordt, is deze groep actief en de andere inactief. Binnen deze groep blijft het toebehorende stuursignaal actief gedurende 2,5 seconden, ongeacht of er voldaan wordt aan de voorwaarden of niet. Om de veiligheid van de gebruiker te garanderen is er binnen elke groep een dead-stop geïmplementeerd. Hierbij komt het platform onmiddellijk tot stilstand zodra er een tegenwerkende kracht waargenomen wordt.

Na deze 2,5 seconden wordt er gedurende 5 seconden rustig vertraagd. Wanneer er tijdens deze 5 seconden terug de initiële voorwaarden geregistreerd worden, hervat het proces en begint opnieuw met de 2,5 seconden.

Na afloop van de 5 seconden of na een dead-stop wordt alles gedurende 0,25 seconden inactief gehouden, dit omdat bij een dead-stop het platform onmiddellijk stopt en dit schommelingen veroorzaakt. In sommige gevallen komen deze schommelingen boven de detectiegrens en reageert het platform als volgt hierop. Deze 0,25 seconden worden daarnaast ook gebruikt om de modules onder het platform te oriënteren onder de gewenste hoek van 45° .

6.3.2 Implementatie in code

De code werd opgesteld op basis van de logica bekomen in het voorgaande deel, deze wordt opgesteld voor de ziekenhuisrobot. De implementatie is onderverdeeld in vier algoritmes, die in dit hoofdstuk worden weergegeven aan de hand van pseudocode. Om een duidelijke link tussen deze algoritmes te leggen is eerst een flowchart opgesteld. Figuur 6.17 toont de algemene structuur van de code en de onderlinge relatie tussen de algoritmes.



Figuur 6.17: Flowchart algoritme iteratie 1

De flowchart laat zien dat de code start met het inlezen van parameters. Deze parameters zijn zoals eerder vermeld, afkomstig uit het PlatformDriver-script. Het versturen naar het ontwikkelde script PlatformController gebeurt door Float64MultiArray. De data heeft hier de vorm van twee arrays, die elk de data van een wielmodule bevatten. Deze array wordt in het algoritme dan

ingelezen en geeft het correcte label aan de parameters. Vervolgens worden de waarden van de parameters gebruikt voor de nodige omvorming naar de verschillende koppels. Het inlezen en omvormen van de parameters wordt behandeld in algoritme 1.

Na de omvorming van de parameters worden deze doorgegeven aan de run-functie. Deze is opgesplitst in twee algoritmes. Het eerste algoritme is de initialisatie en bepaalt welke van de drie interactietypes opgemeten zijn. Het tweede algoritme is één van de drie mogelijke bewegingen. Hier worden nog andere acties uitgevoerd die eigen zijn aan dat soort beweging.

De verschillende bewegingen sturen uiteindelijk de KELO-drives aan, zodat het platform effectief beweegt. Dit wordt gedaan aan de hand van het vierde algoritme, deze ontvangt de snelheidssignalen en stuurt deze als Twist-berichten door naar CmdVel in het PlatformDriver-script. Hier worden de ontvangen snelheden via de Twist-berichten omgevormd naar setpoints en naar de KELO-drives gestuurd over EtherCAT.

Inlezen en verwerking parameters

Algoritme 6.1 beschrijft het inlezen en verwerken van de parameters in de EncoderCallback functie. De ruwe sensorwaarden van de wielen (rotatiesnelheid, stroom, hoekstand pivot en rotatiesnelheid pivot) worden toegewezen aan interne parameters. Vervolgens worden deze waarden omgevormd, volgens eerder bekomen formules en logica, tot koppelwaarden. Om ongewenste fluctuaties en meetfouten op te vangen, wordt gewerkt met een voortschrijdend gemiddelde. Op de aldus berekende koppels wordt in de volgende algoritmes verdergewerkt voor de verdere interpretatie van de gebruikersinteractie.

Algoritme 6.1 Iteratie 1 van de EncoderCallback functie van de ziekenhuisrobot

```

1: Inlezen array afkomstig uit PlatformDriver
2: Array items toekennen aan correcte parameters:
3:  $\omega$ ,  $i_q$  voor L en R voor drive A en B
4:  $\omega_{pivot}$ ,  $\theta_{pivot}$  voor drive A en B
5:
6: Berekening van de koppels:
7: Berekenen koppel uit kwadratuurstroom
8: if Kwadratuurstroom traag daalt bij stilstand then
9:     Werk huidig koppel weg
10: else
11:     Berekenen verwacht koppel
12:     Verschil tussen beide koppels
13: end if
14: if Pivot een grote hoeksnelheid then
15:     Huidig koppel = 0
16: end if
17: Som koppels voor de vier wielen
18: Moving average
19:
20: Berekenen absolute koppel voor drive A en B

```

Verwerking bekomen koppels

Voor de verwerking van de koppels werd een run-functie opgesteld. Deze bestaat uit twee delen; het eerste deel wordt weergegeven in algoritme 6.2. Het doel van dit eerste deel is om op basis van de huidige koppels en encoderwaarden te bepalen welke type van interactie plaatsvindt. Om de keuze van de interactie te vergemakkelijken zijn er eerst nog twee aspecten die behandeld worden. Indien een abrupte snelheidsverandering wordt gedetecteerd, onderbreekt het algoritme de huidige verwerking. Indien de interactie is beëindigd, oriënteert het de modules opnieuw zodat het mogelijk wordt een onderscheid te detecteren tussen zijwaartse en voorwaartse beweging. Daarnaast worden vanuit de waarden van de nulposities van de encoders de grenswaarden berekend, om zo het correcte relatieve beeld van de externe kracht te krijgen. In de rusttoestand wordt dan aan de hand van de encoderwaarden een inschatting gemaakt van de richting van de externe kracht. Het algoritme oriënteert bij een inschatting van een voorwaartse kracht vervolgens de modules in de bijbehorende richting en start een timer. Binnen een tijdsvenster van 2,5 seconden wordt het type beweging geverifieerd (voorwaarts of rotatie), waarna dit wordt doorgegeven aan het volgende algoritme.

Algoritme 6.2 Deel 1 run-functie van de ziekenhuisrobot

```
1: if Interactie is beëindigd then
2:   Oriënteer modules op 45°
3:   Wacht 0,25 seconden
4:   return
5: end if
6: Sla huidige snelheden op
7: Bereken grenswaarden voor encoderposities (A en B)
8: Bepaal teken op basis van oriëntatie
9: if Platform staat stil then
10:   Bepaal richting aan de hand van encoderwaarden
11:   if Koppel en pivot wijzen naar voorwaartse beweging then
12:     Oriënteer modules op 0°
13:     Start timer
14:     Wacht op interactie:
15:     if Koppel en absoluut koppel wijzen op roteren rond centerpunt then
16:       Start rotatie rond centerpunt
17:     else if Koppel wijst op voorwaartse beweging then
18:       Start voorwaartse beweging
19:     else if Timer actief voor 2,5 seconden then
20:       Trigger einde interactie
21:     end if
22:   else if Koppel wijst naar zijwaartse beweging then
23:     Start zijwaartse beweging
24:   end if
25: end if
```

Het tweede deel van de run-functie (algoritme 6.3) verwerkt de toestand wanneer het platform reeds in beweging is. Dit deel bepaalt hoelang de beweging wordt aangehouden, of ze versneld of vertraagd moet worden, en of er nood is aan een correctieve actie zoals het beëindigen van de interactie of draaien. Het versnellen gebeurt indien er een significant verschil is tussen het waargenomen koppel en de detectiegrens voor het koppel. Om niet oneindig te versnellen is er een limiet opgesteld op het dubbele van de initiële snelheid.

Binnen de eerste 2,5 seconden blijft het platform de gekozen beweging volgen, tenzij een tegengestelde kracht wordt waargenomen. Indien dat gebeurt, wordt de interactie abrupt beëindigd. Tijdens deze fase kan ook een overgang naar een draaiende beweging plaatsvinden, dit kan echter enkel voor een voorwaartse beweging. Bij het roteren rond het centerpunt wordt er al gedraaid en doordat er slechts twee KELO-drives zijn, is het onmogelijk om het draaien hier waar te nemen.

Indien de beweging langer dan 2,5 seconden aanhoudt, wordt deze gelijkmatig vertraagd over een periode van 5 seconden. Voor het gemak is er bij het roteren rond het centerpunt gekozen om deze periode in te korten naar 1,25 seconden. Tijdens deze vertraging blijft het platform responsief: een meewerkend koppel kan de timer resetten en de beweging verlengen, terwijl een tegengestelde kracht alsnog een onmiddellijke stop veroorzaakt. Als na 7,5 seconden ($2,5 + 5$) geen significante verandering is waargenomen, wordt de interactie automatisch beëindigd.

Algoritme 6.3 Deel 2 run-functie van de ziekenhuisrobot

```

1: Versnel indien interactie dit toelaat
2: if platform in beweging then
3:   if Minder dan 2,5 seconden verlopen then
4:     Behoud huidige beweging
5:     if Tegengestelde koppel then
6:       Stop
7:       Trigger einde interactie
8:     else if Voorwaartse beweging AND koppel en absoluut koppel wijzen op draaien
       then
9:       if Drive A grootste koppel then
10:        Draaien naar links
11:       else if Drive B grootste koppel then
12:        Draaien naar rechts
13:       end if
14:     end if
15:   else if Meer als 2,5 seconden verlopen then
16:     Vertraag huidige beweging gedurende 5 seconden
17:     if Tegengestelde koppel then
18:       Stop
19:       Trigger einde interactie
20:     else if Meewerkend koppel then
21:       Reset timer
22:     end if
23:   else if Meer als 7,5 seconden verlopen then
24:     Trigger einde interactie
25:   end if
26: end if
27: Update timer

```

Uitsturen snelheden

De bovenstaande run-functie stuurt de gewenste snelheden door naar een afzonderlijke publicatiefunctie, weergegeven in algoritme 6.4. Deze functie heeft als belangrijkste doel het veilig verzenden van snelheidscommando's naar de KELO-drives.

Voor het verzenden wordt gecontroleerd of de snelheden binnen vooraf ingestelde grenzen blijven. Dit is vooral van belang tijdens de testfase of bij onverwachte inputs, om te vermijden dat het platform gevaarlijk gedrag vertoont bij foutieve snelheidswaarden. Indien een snelheid hoger is dan de toegestane maximumwaarde, worden alle snelheden gereset naar nul.

De snelheden die effectief verstuurd worden, worden tevens opgeslagen. Deze opgeslagen waarden worden gebruikt in algoritme 6.2, bij de detectie van hoge snelheidsveranderingen. Tot slot worden de snelheden verstuurd via ROS Twist-berichten naar het PlatformDriver-script, dat deze vertaalt naar setpoints voor de wielmodules.

Algoritme 6.4 Publicatiefunctie snelheid

- 1: Inlezen uit te sturen snelheden
 - 2: **if** Een snelheid groter dan maximaal toegelaten snelheid **then**
 - 3: Reset snelheden naar 0
 - 4: **end if**
 - 5:
 - 6: Sla huidige doorgestuurde snelheid op
 - 7:
 - 8: Publiceer snelheden via Twist-berichten
-

Hoofdstuk 7

Aansturing mobiel platform met vier KELO-drives

Met de overgang van het prototypeplatform met twee aangedreven wielen naar een volwaardig mobiel platform met vier KELO-drives, waarop een collaboratieve robotarm gemonteerd wordt, is er nood aan een meer robuuste, flexibele en schaalbare aansturing. De eerder ontwikkelde codebase, afgestemd op een lichtere constructie en tweewielige aandrijving, voldoet niet langer aan de eisen die het nieuwe platform theoretisch zou stellen. Om die reden werd het volledige algoritme herwerkt, met een focus op schaalbaarheid (naar 2, 3 of 4 wielmodules), uitbreidbaarheid en dynamische configuratie. De nieuwe softwareversie vormt de brug tussen een proof-of-concept en een inzetbaar sturingssysteem voor een mobiel platform in een echte werkomgeving. Hiervoor worden de parameters makkelijker aanpasbaar en kalibreerbaar gemaakt zodat deze niet bij elk platform manueel gezocht moeten en deze in de code aangepast moeten. Daarnaast wordt er ook in het platformDriver-script een nieuwe functie aangemaakt, deze elimineert de nood aan een tussenfase voor het controleren naar voorwaartse beweging en maakt het platform in het algemeen makkelijker bestuurbaar.

7.1 Bijkomende obstakels en functionaliteiten

De eerste en voornaamste stap bij het overschakelen van een platform met twee aangedreven wielen naar een met vier aangedreven wielen, is het zorgvuldig in kaart brengen van de bijkomende obstakels die deze overgang theoretisch zou kunnen brengen. Een eerste voor de hand liggend verschil is uiteraard het aantal wielmodules, maar vooral de bijhorende kinematische lay-out zorgt voor fundamentele verschillen. Waar bij het platform met twee wielen de wielmodules op één lijn liggen, vormt het geheel bij drie of meer wielmodules een vlak. Dit leidt tot een overgang van een eendimensionale naar een tweedimensionale aanpak en heeft impact op meerdere niveaus:

- De koppelberekening wordt aanzienlijk complexer; het simpelweg optellen van koppels volstaat niet langer.
- De groepering en oriëntatie van de wielen moet expliciet beheerd worden.
- De impact van de positie van elke drive t.o.v. het rotatiecentrum wordt aanzienlijk groter bij krachtmetingen.

Naast deze structurele verschillen komen ook andere, minder zichtbare obstakels naar voren. Zo is het gewicht van het platform, en belangrijker nog, de gewichtsverdeling, sterk variabel door de toevoeging van de collaboratieve robotarm en eventuele belasting. Dit zorgt ervoor dat er standaard hogere koppels optreden, waardoor de detectiegrenzen moeten aangepast worden om foutposities te vermijden.

De bestaande detectiemethode van iteratie 1 is ook onvoldoende. Deze methode stuurt kortstondig een hoeksnelheid uit naar de drives om ze kunstmatig in een gewenste hoek te oriënteren (ongeveer 70°), zodat onderscheid kan gemaakt worden tussen voorwaartse en zijwaartse krachten. Deze aanpak heeft meerdere nadelen:

- De hoekverandering is relatief klein, waardoor het onderscheid tussen richtingen onvoldoende uitgesproken is.
- Het snelheidssignaal zorgt voor een korte, ongewenste verplaatsing van het platform.
- Er is een aparte tussenfase nodig om de richting te bevestigen, wat de besturing complexer maakt.
- De bekomen hoek is afhankelijk van de kinematische lay-out.

Een eerste poging om deze problemen op te lossen is het introduceren van een *bias*, deze wordt vermenigvuldigd met het signaal van de wielmodule om zo voor- en zijwaartse krachten te egaliseren. Deze bias is afhankelijk van de actuele oriëntatie van elke drive. In praktijk leidt dit echter tot inconsistent gedrag, doordat de hoekinstellingen via de bestaande methode met snelheidsimpulsen niet exact reproduceerbaar zijn. De wielmodules bevinden zich dan nog occasioneel in een positie nabij negentig graden, waardoor een zijwaarts of voorwaarts koppel niet gemeten wordt.

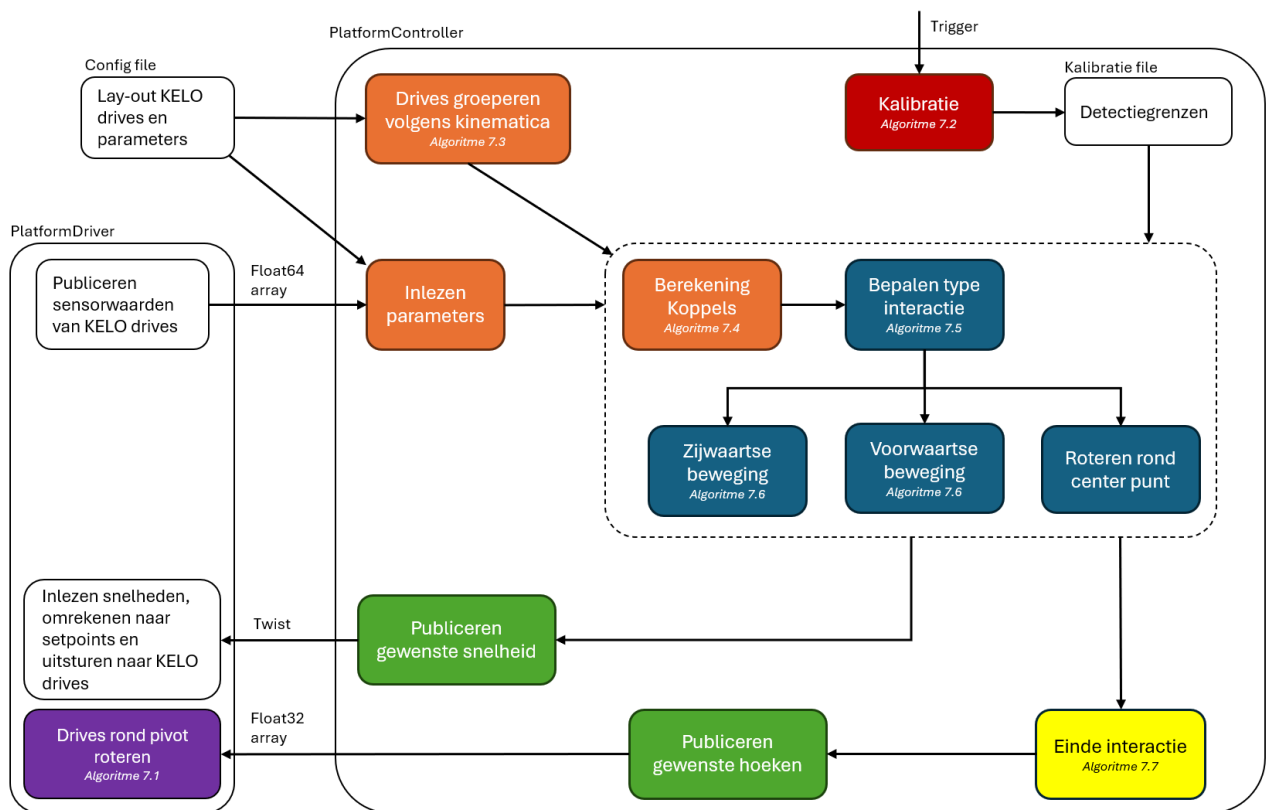
Om dit op te lossen is een nieuwe functionaliteit geïntroduceerd in het bestaande platformDriver-script. Deze functie laat het toe om elke wielmodule afzonderlijk en rechtstreeks een gewenste oriëntatiehoek mee te geven, onafhankelijk van snelheidsimpulsen. Initieel is gekozen om alle wielmodules een hoek van 45° te geven, in een poging om zowel voorwaartse als zijwaartse detectie tegelijk te behouden. Het blijkt echter snel dat er een gunstigere oriëntatie is om zijwaarts en voorwaarts volledig te scheiden: de wielmodules worden onder 0° (voorwaarts) en onder 90° (zijwaarts) gezet. Hierdoor wordt de koppeldetectie per richting expliciet en eenduidig, wat zowel de nauwkeurigheid als de bestuurbaarheid van het platform bevordert.

Deze expliciete scheiding van richtingen maakt ook de weg vrij voor een volgende, context-specifieke uitbreiding: het zijwaarts volgen van een muur bij contactdetectie. Hierbij gaat het platform, wanneer het zich dicht bij een muur bevindt en een kracht richting die muur detecteert, automatisch zijwaarts bewegen om zich uit te lijnen met de muur. Deze functionaliteit is belangrijk in het kader van interactie met vaste werkstations in een industriële omgeving. De logica en basiscode hiervoor zijn reeds voorzien, maar de effectieve implementatie volgt in toekomstig werk.

7.2 Aanpassingen aan het algoritme

De uitbreiding van het platform van twee naar vier aangedreven wielmodules impliceert meer dan een lineaire schaalvergroting. De overgang naar een tweedimensionale kinematische structuur vereist een grondige herwerking van zowel de onderliggende bewegingslogica als van de manier waarop interacties worden geïnterpreteerd en uitgevoerd. Daarbij worden verschillende aspecten van de bestaande architectuur aangepast: de structuur van de initialisatie, de interpretatie van krachten en koppels, het selectieproces voor interactietypes, alsook de controle over de uitvoering en beëindiging van acties.

Om het verschil met de eerste iteratie van de code makkelijk weer te geven, is ook weer een flowchart opgesteld. Deze bevat nog vergelijkbare basiselementen uit de eerste iteratie, maar bezit ook al extra uitbreidingen om compatibel te zijn met vier wielmodules. Figuur 7.1 toont de uitgebreide flowchart met de onderlinge relaties tussen de verschillende algoritmes.



Figuur 7.1: Overzicht van de interactie- en aansturingslogica van het platform

De flowchart van de tweede iteratie laat zien dat een groot deel van de logica behouden is. Het algemene verloop van het ontvangen en inlezen van de parameters, berekening van de koppels, bepalen welk type interactie er gemeten wordt, de corresponderende beweging uitvoeren en het sturen van de snelheden is in essentie gelijk gebleven. Inhoudelijk zijn deze algoritmes wel veranderd.

Een eerste uitbreiding is het algoritme om de wielmodules rond hun zwenkmechanisme te roteren. Dit maakt dat de oriëntatie van elke individuele module gekozen kan worden. Hieruit volgt ook het nieuwe algoritme om deze oriëntatie te sturen. Vervolgens wordt de configuratie-file gebruikt om eenvoudiger belangrijke parameters in te stellen. De kinematica van het platform wordt hier ook ingesteld. Het volgende nieuwe algoritme gebruikt deze kinematica om de wielmodules te groeperen en andere eigenschappen toe te kennen. Ten slotte is een algoritme voor kalibratie aangemaakt om zo de detectiegrenzen in real-time te berekenen en bij te stellen.

7.2.1 Toevoegingen aan bestaande driver code

Een kernonderdeel van de herwerking bestaat uit het aanpakken van de huidige manier waarop de wielmodules een gunstige oriëntatie aannemen om het koppel te meten. Waar in de vorige iteratie alle wielmodules georiënteerd worden door een korte hoeksnelheid aan te sturen, verandert dit naar een nieuwe stuurfunctie die toelaat naar elke wielmodule individueel een hoek te sturen die aangenomen dient te worden. Als eerste stap wordt de veiligheid van de KELO 100 drives gegarandeerd door het enkel mogelijk te maken dat er slechts één van de twee stuurfuncties tegelijk gebruikt kan worden. Daarnaast worden de bestaande veiligheden uit de andere stuurfunctie overgenomen, dit omvat het voorkomen dat er te kleine ($< 0,01$) of te grote (> 35) waarden voor de setpoints aangestuurd worden. Om de waarden van de setpoints te bepalen is een eenvoudige KP-regelaar geïmplementeerd.

Deze regelaar werkt op basis van de hoekfout θ_{fout} tussen de gewenste hoek θ_{doel} en de huidige hoek θ_{huidig} van een wielmodule:

$$\theta_{\text{fout}} = \theta_{\text{doel}} - \theta_{\text{huidig}} \quad (7.1)$$

Vervolgens wordt deze fout omgezet naar een roterende beweging door de setpoints in tegengestelde richting aan te sturen. Hierdoor roteert de wielmodule rond zijn eigen as zonder translatie van het platform. De formule waarmee dit wordt omgezet naar de setpoints voor het linker- en rechterwiel van een wielmodule is als volgt:

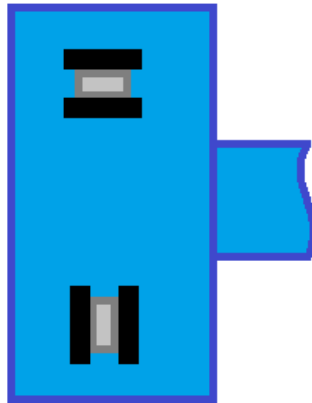
$$\omega_{\text{links}} = -K_P \cdot \theta_{\text{fout}}, \quad \omega_{\text{rechts}} = K_P \cdot \theta_{\text{fout}} \quad (7.2)$$

Waarbij K_P een proportionele versterkingsfactor is die afgestemd wordt op basis van de gewenste responssnelheid. Uit testen is ondervonden dat de waarden van de proportionele versterkingsfactor het best tussen twee en vier ligt, om niet te traag maar ook niet te hevig te reageren. Als laatste is er een tolerantie van vier graden geïmplementeerd op de gewenste hoek zodanig dat er geen signaal meer uitgestuurd wordt naar de wielmodule indien de huidige hoek hierbinnen valt. De functie is weergegeven in algoritme 7.1.

Algoritme 7.1 Wielmodules rond hun centrale as draaien

```
1: Invoer: Doelhoeken per wiel targetPivotAngles
2: Constanten:  $K_P = 4.0$ , tolerantie = 0.08,  $setpoint_{\min} = 0.01$ ,  $setpoint_{\max} = 35.0$ 
3: for elke wielmodule do
4:   Lees huidige hoek  $hoek_{\text{huidig}}$  van encoder
5:   Corrigeer gewenste hoek  $hoek_{\text{doel}}$ 
6:   Bereken kortste hoekfout:  $f_{\text{out}} = hoek_{\text{doel}} - hoek_{\text{huidig}}$ 
7:   Bepaal hoeksnelheid:  $v = K_P \cdot f_{\text{out}}$ 
8:   Beperk  $v$  tot binnen  $[-35.0, 35.0]$ 
9:   if  $|v| < 0.01$  of  $|f_{\text{out}}| < 0.08$  then
10:      $v \leftarrow 0$ 
11:   end if
12:   Stel  $setpoint_1 = -v$  en  $setpoint_2 = -v$ 
13:   Verstuur setpoints naar wielmodule
14: end for
15: Controle: Zijn alle hoeken binnen tolerantie?
16: if voor elk wiel  $|f_{\text{out}}| < 0.08$  then
17:   Zet pivotingActive op false
18: end if
```

Door het toevoegen van dit algoritme is de stand van de wielmodules accurater dan met de oudere methode op basis van snelheidsimpulsen. Dit is belangrijk bij een veranderlijk aantal wielmodules of andere kinematische configuraties. Zoals eerder vermeld is er gekozen om de wielen onder 0° en 90° te oriënteren bij stilstand, een voorstelling hiervan is weergegeven op figuur 7.2. Deze oriëntaties zorgen er voor dat zijwaartse en voorwaartse krachten duidelijk en onafhankelijk gedetecteerd worden.



Figuur 7.2: Oriëntaties wielmodules bij stilstand voor iteratie 2

7.2.2 Parameters

Om de code in de toekomst makkelijker aanpasbaar te maken zonder in de broncode op zoek te moeten gaan naar parameters, is een configuratiefile opgesteld. Deze maakt het mogelijk om volgende parameters in te stellen:

- **Maximale snelheid:** Bepaalt de bovenlimiet van de wielsnelheden.
- **Initiele snelheden:** Startwaarden voor zowel lineaire als hoeksnelheden.

- **detectiegrens factoren:** Factoren voor de detectiegrenzen die bepalen wanneer een kracht groot genoeg is om actie te ondernemen. Door deze factor te verlagen wordt het platform gevoeliger voor de dead-stop of detecteert het koppels sneller. Deze factor vergroten heeft dan het omgekeerde effect.
- **Koppeldetectie aan/uit:** Maakt het mogelijk om de sensorwaarden te loggen en analyseren zonder dat het platform fysiek reageert.
- **Logging:** Met bestandsnaam. Laat analyse toe van ruwe sensorwaarden.
- **LiDAR-functionaliteit:** Kan geactiveerd worden voor objectdetectie of muurvolgning.

De parameters worden bij opstart automatisch ingeladen. Door de code op deze manier te parametriseren, wordt debugging en verfijning van de prestaties aanzienlijk eenvoudiger, vooral in testomgevingen.

Naast de handmatig instelbare parameters via de configuratiefile, is er ook een automatische kalibratiefunctie toegevoegd. Deze functie bepaalt een aantal detectiegrenzen die nodig zijn voor een correcte interpretatie van de gemeten koppels. De kalibratie omvat vier soorten grenswaarden:

- **Bovenste detectiegrens:** minimumwaarde voor positieve koppels.
- **Onderste detectiegrens:** minimumwaarde voor negatieve koppels.
- **Detectiegrens absoluut koppel:** maximale waarde van het absolute koppel.
- **Detectiegrens percentageverschil:** percentageverschil tussen de twee voorwaarts gerichte drives tijdens rotatie (om een draaibeweging te bevestigen).

De kalibratie verloopt in drie fasen:

1. Het platform rijdt gedurende 2 seconden voorwaarts.
2. Daarna stopt het 1 seconde volledig.
3. Tot slot rijdt het 2 seconden achterwaarts.

Tijdens de achterwaartse fase worden de koppelwaarden opgeslagen. De laagste en hoogste waarden binnen deze periodes worden daarna geselecteerd, en als grenzen voor detectie ingesteld. Dit voorkomt foutposities bij normaal rijgedrag, zoals lichte schommelingen of inertie krachten bij het versnellen of stoppen.

Voor het bepalen van de absolute koppelgrens en het percentageverschil tussen de aandrijvingen, wordt specifiek gekeken naar het interval tussen seconde 0.5 en seconde 1.5 van de achterwaartse beweging. In deze periode is de beweging voldoende stabiel, en kunnen realistische, betrouwbare waarden geregistreerd worden. Waarnemingen buiten dit venster geven vaak onnatuurlijke pieken, bijvoorbeeld bij het aanzetten of afremmen, en worden dus genegeerd. Bovendien is er vóór seconde 0.5 sowieso nog geen fysieke respons mogelijk van het platform, waardoor metingen in die fase irrelevant zijn.

Alle gekalibreerde waarden worden na afloop van de procedure automatisch opgeslagen in een aparte `.yaml`-configuratiefile. Deze wordt bij elke opstart van de code opnieuw ingeladen, waardoor er geen nood is aan herkalibratie bij elke heropstart. Indien gewenst kan de kalibratie handmatig opnieuw geactiveerd worden via een ROS-trigger. Algoritme 7.2 geeft de pseudocode van het verloop van de kalibratie.

Algoritme 7.2 Kalibratieprocedure van koppel- en draaidetectiegrenzen

```
1: reset alle grenzen bij eerste call
2: if  $t \leq 2s$  then
3:   SETVELOCITYCOMMAND(0.1, 0, 0) ▷ Voorwaarts rijden
4: else if  $t \leq 3s$  then
5:   SETVELOCITYCOMMAND(0, 0, 0) ▷ Stop
6: else if  $t \leq 5s$  then
7:   SETVELOCITYCOMMAND(-0.1, 0, 0) ▷ Achterwaarts rijden
8:   for all groepen in {voor, achter, links, rechts} do
9:     meet koppel
10:     $max\_koppel \leftarrow \max(max\_torque, koppel)$ 
11:     $min\_koppel \leftarrow \min(min\_koppel, koppel)$ 
12:    verzamel absolute koppel
13:    if  $t \in [3.5s, 4.5s]$  then
14:      if  $som > max\_draai\_koppel$  then
15:         $max\_draai\_koppel \leftarrow som$ 
16:      end if
17:       $percentage \leftarrow c_1 / som$ 
18:      if  $percentage > max\_draai\_percentage$  and  $percentage < 0.8$  then
19:         $max\_draai\_percentage \leftarrow percentage$ 
20:      end if
21:    end if
22:  end for
23: else if  $t \leq T_{\text{totaal}}$  then
24:   SETVELOCITYCOMMAND(0, 0, 0) ▷ Finale stop
25: else
26:   schrijf waarden weg naar calibration.yaml:
27:   zet platformstatus naar POST_INTERACTION
28: end if
```

7.2.3 Structuurwijzigingen en uitbreidingen

Initialisatie

De code initialiseert het platform op dynamische wijze, met ondersteuning voor configuraties met twee, drie of vier aangedreven wielmodules. Dit vereist fundamentele wijzigingen ten opzichte van de voorgaande iteratie, die uitgaat van een vaste structuur. De initialisatie bestaat uit drie hoofdstappen: zijdegroepering, oriëntatietoekenning en biascompensatie. Deze stappen zijn elk weergegeven met pseudocode in algoritmes 7.3.1, 7.3.2 en 7.3.3.

In een eerste stap groepeer algoritme 7.3.1 elke wielmodule op basis van de bijhorende coördinaten in de configuratie. Elke module wordt toegewezen aan twee zijdes van het platform: voor, achter, links of rechts. Deze groepering gebeurt door de x- en y-posities van de wielmodule relatief te evalueren ten opzichte van het middelpunt. Een wielmodule die bijvoorbeeld links en vooraan op het platform is geplaatst, behoort dan tot zowel de links- als de voor-groep.

Algoritme 7.3.1 Groeperen van wielmodules per zijde

```
1: Leeg de lijsten: voor, achter, links, rechts
2: for elke wielmodule  $i$  do
3:   Lees positie  $(x_i, y_i)$ 
4:   if  $x_i > 0$  then
5:     Voeg  $i$  toe aan groep voor
6:   else if  $x_i < 0$  then
7:     Voeg  $i$  toe aan groep achter
8:   end if
9:   if  $y_i > 0$  then
10:    Voeg  $i$  toe aan groep links
11:   else if  $y_i < 0$  then
12:    Voeg  $i$  toe aan groep rechts
13:   end if
14: end for
```

In een tweede stap kent algoritme 7.3.2 elke module een standaardoriëntatie toe, specifiek voor gebruik bij stilstand. In het geval van twee wielmodules wordt één module georiënteerd in voorwaartse richting, en de andere in zijwaartse richting. Bij drie of meer modules wordt een dambordpatroon toegepast, waarbij oriëntaties van 0° en 90° afwisselend worden toegekend. Dit zorgt voor een gebalanceerde verdeling van koppeldetectie in de voorwaartse als zijwaartse richting.

Algoritme 7.3.2 Bepalen van standaardoriëntatie per wielmodule

```
1: Stel  $hoek_{voorwaarts} \leftarrow 0$ ,  $hoek_{zijwaarts} \leftarrow \frac{\pi}{2}$ 
2: if aantal wielmodules = 2 then
3:   Vergelijk  $y$ -posities
4:   if verschil in  $y$  groot genoeg then
5:     Lagere  $y$ : voorwaarts, hogere  $y$ : zijwaarts
6:   else
7:     Vergelijk  $x$ -posities
8:     Rechter: voorwaarts, linker: zijwaarts
9:   end if
10:  Ken overeenkomstige hoeken toe aan beide wielmodules
11: else
12:  for elke wielmodule  $i$  do
13:    Bepaal hoek op basis van dambordpatroon
14:    Wijs  $hoek_{voorwaarts}$  of  $hoek_{zijwaarts}$  toe aan  $i$ 
15:  end for
16: end if
```

De laatste stap omvat een correctie voor de onbalans die optreedt bij platforms met twee of drie wielmodules. Indien slechts één module op een bepaalde zijde gepositioneerd is, en deze niet loodrecht gericht staat op die zijde, treedt een meetvervorming op. De gemeten kracht verschilt dan naargelang de afstand tot de krachtinwerking. Om dit effect te compenseren, berekent algoritme 7.3.3 een bias-factor. Deze wordt toegepast op de gemeten kracht van de betrokken module, zodat de interpretatie van externe krachten symmetrisch blijft over het volledige platform.

Algoritme 7.3.3 Berekening van de bias-factor

```

1: for elke zijde (voor, achter, links, rechts) do
2:   if de zijde bevat exact één wielmodule then
3:     if de oriëntatie van de module stemt overeen met de zijde then
4:       Stel bias in op 1.0
5:     else
6:       Stel bias in op 0.6                                ▷ (compensatie voor meetvervorming)
7:     end if
8:   end if
9: end for

```

Deze drie stappen vervangen het vaste gedrag uit de vorige iteratie, en maken de code compatibel met verschillende, fysiek asymmetrische configuraties. De koppeldetectie, actieaansturing en oriëntatielogica worden zo vanaf de initialisatie afgestemd op de kinematica van het platform.

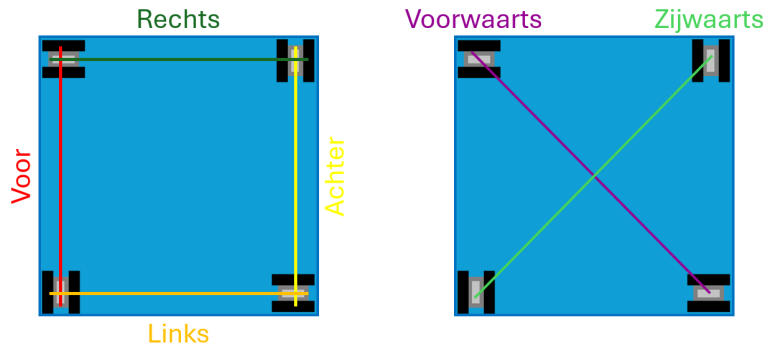
Operatie

De centrale aanstuurfunctie van het platform is grondig herzien om compatibel te zijn met meerdere wielmodules en een tweedimensionale detectiestructuur. Net zoals in de vorige iteratie start de functie bij elke lus met het inlezen van gegevens van de wielmodules. In tegenstelling tot de vorige iteratie worden meerdere koppels berekend, verdeeld over twee assen (voorwaarts en zijwaarts) en vier zijdes (voor, achter, links, rechts). Elke wielmodule draagt bij tot beide structuren: enerzijds op basis van de bijhorende oriëntatie in stilstand (0° of 90°), anderzijds op basis van de fysieke positie op het platform (via eerder gedefinieerde groepering). Algoritme 7.4 omvat de herwerking van de EncoderCallback-functie. De berekening van het koppel is hetzelfde gebleven, alsook de constanten die de berekening ondersteunen.

Algoritme 7.4 Verwerken van encoderdata en toekennen van de koppels

```
1: Invoer: Encoder- en stroomdata per wielmodule
2: Uitvoer: Geaggregeerde koppels: voorwaarts, zijwaarts, links, rechts, voor, achter
3:
4: Initialiseer alle koppelwaarden op nul
5: for elke wielmodule  $i$  do
6:   Lees snelheden, stromen en encoderwaarden uit het ontvangen bericht
7:   Bereken  $torque_L$  en  $torque_R$  op basis van snelheden, stromen en vorige waarde
8:   Bepaal nettokoppel:  $diff = torque_R - torque_L$ 
9:   Sla absolute stroomsterkte op:  $|torque_L| + |torque_R|$ 
10:
11:   if hoek wielmodule = 0 then
12:     Tel diff op bij totaal voorwaarts koppel
13:   else if hoek wielmodule =  $\frac{\pi}{2}$  then
14:     Tel diff op bij totaal zijwaarts koppel
15:   end if
16:
17:   if  $y_i < 0$  then
18:     Tel diff op bij koppel links
19:   else if  $y_i > 0$  then
20:     Tel diff op bij koppel rechts
21:   end if
22:   if  $x_i > 0$  then
23:     Tel diff op bij koppel voor
24:   else if  $x_i < 0$  then
25:     Tel diff op bij koppel achter
26:   end if
27: end for
28:
29: Sla laatste 10 waarden van elke zijde op (voor voortschrijdend gemiddelde)
30: Bereken gemiddeld koppel per zijde
31: Sla deze op als  $torque.\{left, right, front, back\}$ 
32: Sla  $total\_forward\_torque$  en  $total\_sideways\_torque$  op in platformdata
```

In tegenstelling tot de koppels van de vier zijdes, wordt op de voorwaartse en zijwaartse koppels geen voortschrijdend gemiddelde toegepast, zodat er sneller en accurater gereageerd wordt. Alle waarden worden net zoals voorgaande iteratie gelogd voor latere analyse. Een betere visuele voorstelling van de zes verschillende koppels is weergegeven in figuur 7.3 voor het platform met vier wielmodules.



Figuur 7.3: Voorstelling van de zes koppels op het platform met vier wielmodules

In bovenstaande figuur is het platform met vier wielmodules bekeken langs onder en staan de wielmodules in de standaardconfiguratie. Deze figuur laat zien hoe elke module tot twee zijdes behoort en elke zijde ook zo een voorwaarts en een zijwaarts georiënteerde module bevat.

Bij het bepalen van de interactie vertrekt de code vanuit stilstand. In tegenstelling tot de vorige iteratie, waarin slechts één gecombineerd koppel wordt gebruikt en de interpretatie van richtingskrachten afhankelijk is van de rotatie van het zwenkmechanisme, werkt de vernieuwde algoritme 7.5.1 met de twee afzonderlijke koppelwaarden. Zodra één van beide koppels buiten de detectiegrenzen valt, wordt de bijbehorende beweging geactiveerd. De bewegingsrichting wordt bepaald op basis van het teken van het koppel: positief of negatief wijst respectievelijk op een kracht naar voren of naar achteren (voorwaarts), of naar links of rechts (zijwaarts).

Algoritme 7.5.1 Detectie van interactie vanuit stilstand

```

1: Zet versnelling op nul en startparameters op standaardwaarden
2: Lees  $f_{koppel}$  en  $s_{koppel}$  (voorwaarts en zijwaarts koppel)
3:
4: Sterke detectie:
5:    $sterk_{voor} \leftarrow f_{koppel}$  ▷ buiten boven-/ondergrens (met bias)
6:    $sterk_{zij} \leftarrow s_{koppel}$  ▷ buiten boven-/ondergrens (met bias)
7:
8: Zwakke detectie:
9:    $zwak_{voor} \leftarrow f_{koppel}$  ▷ buiten halve grens
10:   $zwak_{zij} \leftarrow s_{koppel}$  ▷ buiten halve grens
11:
12: if  $zwak_{voor}$  én  $zwak_{zij}$  en niet  $sterk_{voor}$  en niet  $sterk_{zij}$  then
13:   Zet toestand op ROTATIE
14:   Bepaal draairichting op basis van teken van  $f_{koppel}$ 
15: else if  $sterk_{voor}$  then
16:   Zet toestand op VOORWAARTSE BEWEGING
17:   Bepaal richting (vooruit/achteruit) op basis van teken van  $f_{koppel}$ 
18:   Bepaal dominante zijde via stroommetingen
19:   SETVELOCITYCOMMAND( $v_{init} \cdot richting$ , 0, 0) ▷ x-richting
20: else if  $sterk_{zij}$  then
21:   Zet toestand op ZIJWAARTSE BEWEGING
22:   Bepaal richting (links/rechts) op basis van teken van  $s_{koppel}$ 
23:   Bepaal dominante zijde via stroommetingen
24:   SETVELOCITYCOMMAND(0,  $v_{init} \cdot richting$ , 0) ▷ y-richting
25: end if

```

Rotatie rond het middelpunt van het platform vereist dat beide koppels zich tegelijk binnen een specifiek bereik bevinden: tussen de helft en de volledige waarde van hun respectieve detectiegrens. Deze drempel voorkomt dat kleine krachten per ongeluk een draaibeweging activeren. Het gebruik van het absoluut koppel is hierdoor overbodig. Het enkel gebruik van beide koppels geeft voldoende onderscheid tussen de drie mogelijke bewegingen. Hierdoor zijn geen extra parameters nodig.

Na het selecteren van een beweging op basis van de gedetecteerde interactie ontstaat een bijkomende uitdaging: bepalen welk koppel gebruikt wordt om de beweging aan te sturen. In tegenstelling tot de voorgaande versie, waarin slechts één koppel wordt beschouwd, bestaan er nu meerdere potentiële kandidaten.

Algoritme 7.5.2 selecteert daarom uitsluitend het koppel van zijdes die uit twee wielmodules bestaan. Dit levert betrouwbaardere en stabielere signalen op, aangezien ruis of afwijkingen op één individuele module minder impact hebben op het totaal. Verdere bespreking en analyse over het koppel van een individuele module volgt na het testen met het vierkant parcours in deel 7.4.1. Bij een platform met twee wielmodules is het vanzelfsprekend welke zijde geselecteerd wordt, maar bij vier wielmodules dient de zijde geïdentificeerd te worden waarop effectief kracht uitgeoefend wordt. Het is niet mogelijk om dit enkel af te leiden uit het teken van het globale voorwaartse of zijwaartse koppel alleen. Deze richtingen leveren zowel bij duw- als trekkrachten identieke signalen op. In de situatie van een platform met vier wielmodules dat in de voorwaartse richting wordt geduwd, gaat eerst bepaald worden welke van de twee wielmodules met de voorwaartse oriëntatie het grootste absolute koppel meet. Daarna wordt de configuratie gebruikt om deze wielmodule aan de correcte zijde te linken.

Algoritme 7.5.2 Bepalen van dominante zijde op basis van koppels en configuratie

```

1: Invoer: Modus: voorwaarts (true) of zijwaarts (false)
2: Zoek de wielmodule met het hoogste absolute koppel en correcte oriëntatie
3: for elke wielmodule i do
4:     if hoek van wielmodule is gealigneerd met bewegingsmodus then
5:         if absolute koppel > huidige maximum then
6:             Bewaar stroom en index van deze wielmodule als dominant
7:         end if
8:     end if
9: end for
10:
11: if modus is voorwaarts then
12:     if dominante wielmodule aan achterzijde then
13:         Interactie van achterzijde
14:     else
15:         Interactie van voorzijde
16:     end if
17: else
18:     if dominante wielmodule aan rechterzijde then
19:         Interactie van rechterzijde
20:     else
21:         Interactie van linkerzijde
22:     end if
23: end if

```

Na detectie van een interactie wordt daarom eerst de dominante zijde bepaald: dit is de zijde met twee wielmodules die de hoogste absolute koppelwaarde meet. Enkel het koppel van deze zijde wordt vervolgens gebruikt om de beweging uit te voeren. Dit zorgt ervoor dat krachten correct geïnterpreteerd worden, ongeacht de oriëntatie van het platform of de richting van de krachtinwerking.

De fundamentele werking van de voorwaartse beweging is grotendeels hetzelfde gebleven. Het draaien tijdens deze beweging is echter herwerkt om beter rekening te houden met vier wielmodules. In het nieuwe algoritme 7.6 is draaien ook geïntegreerd in de zijwaartse interactielogica. Indien de desbetreffende zijde over twee wielmodules beschikt, is het mogelijk om draaien te detecteren. Gelijk de voorgaande iteratie wordt er gedraaid door het procentueel verschil in koppel tussen de wielmodules. Op de betrokken zijde wordt er bepaald welke wielmodule links en rechts op de zijde ligt. Afhankelijk van deze ligging wordt er dan tijdens de beweging linksom of rechtsom gedraaid.

Algoritme 7.6 Detectie van rotatie tijdens voorwaartse of zijwaartse beweging

```

1: if bewegingskoppel < 75% van ondergrens then
2:   if exact twee drives op actieve zijde then
3:     Wijs drive links/rechts toe op basis van y-coördinaat
4:     Bereken som van absolute koppels
5:     if som > absolute koppelgrens then
6:       Bereken koppelpercentages links/rechts
7:       if links > percentagegrens then
8:         Zet draairichting op met de klok mee
9:       else if rechts > percentagegrens then
10:        Zet draairichting op tegen de klok in
11:      end if
12:    end if
13:  else
14:    Te weinig data voor rotatiebeslissing
15:    Zet draairichting op geen
16:  end if
17: end if
18:
19: Instellen van draaisnelheid:
20: if tijd > 0,5s then
21:   Draairichting reset naar geen
22: else if draairichting = met de klok mee then
23:   Zet hoeksnelheid positief, reduceer lineaire snelheid
24: else if draairichting = tegen de klok in then
25:   Zet hoeksnelheid negatief, reduceer lineaire snelheid
26: end if

```

Gelijk aan de voorgaande iteratie is er een aparte draaimodus betreft het roteren rond het centrale punt van het platform. Aangezien deze rotatie slechts twee mogelijke richtingen heeft (met of tegen de klok in), is de specifieke zijde waarop de kracht wordt uitgeoefend niet van belang. Er wordt enkel gekeken naar de zijdes met twee wielmodules, en daaruit wordt op basis van de laagste en hoogste koppelwaarden beslist of de rotatie voortgezet wordt of beëindigd.

Een belangrijke aanpassing betreft de einde interactie functie. Bij het stoppen van een actie worden expliciet de standaardoriëntaties naar elke wielmodule verzonden. De overgang naar een nieuwe interactie wordt pas toegestaan wanneer alle modules hun standaard hoeken bereikt hebben. Deze controle zorgt er voor dat het platform telkens de krachten op dezelfde manier kan interpreteren. Tot slot is ook de versnellings functie aangepast. Bij detectie van een voldoende groot koppel wordt nu niet enkel de snelheid aangepast, maar ook de tijd dat deze snelheid aangehouden wordt. Deze is weergegeven met algoritme 7.7.

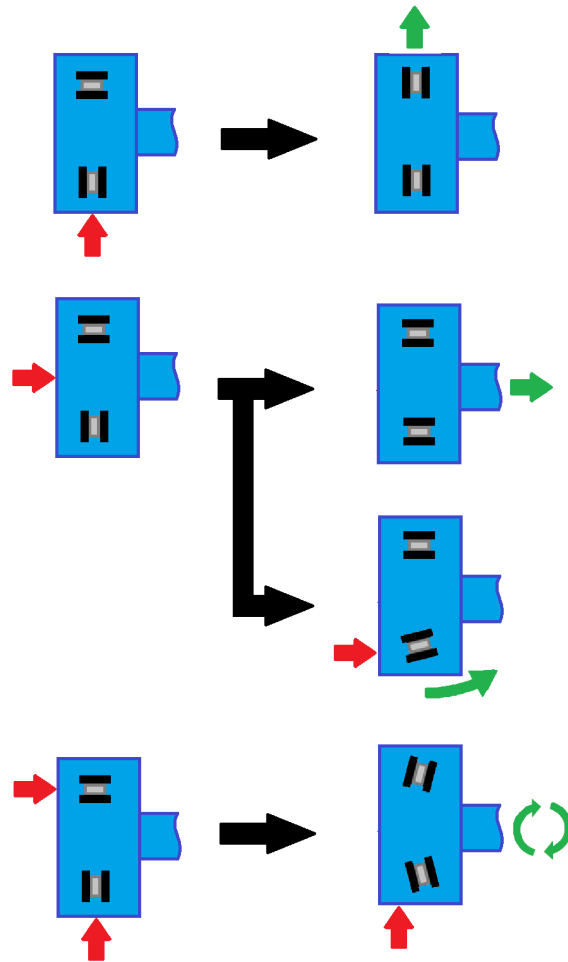
Algoritme 7.7 Heroriëntatie na einde interactie

```

1: Controleer of alle wielen binnen  $5^\circ$  van hun verwachte hoek staan
2: for elke wielmodule  $i$  do
3:   Corrigeer actuele hoek (encoder_positie + offset) naar  $[-\pi, \pi]$ 
4:   Bereken hoekverschil t.o.v. verwachte hoek
5:   if hoekverschil  $> 5^\circ$  then
6:     Niet alle wielen juist georiënteerd
7:     Brek lus af
8:   end if
9: end for
10:
11: if tijd  $> 1,3s$  then
12:   Stop beweging, zet toestand naar IDLE
13: else if tijd  $> 0,7s$  en alle wielen juist georiënteerd then
14:   Stop beweging (wielen juist uitgelijnd)
15: else if tijd  $> 0,2s$  then
16:   Stuur hoekcommando's naar de wielmodules
17: else
18:   Behoud snelheid op nul
19: end if

```

Voor de code van de tweede iteratie is het ook mogelijk om een gebruiksaanwijzing op te stellen net zoals bij de eerste iteratie. Figuur 7.4 geeft deze gebruiksaanwijzing weer. Door het overbodig maken van een tussenfase is deze eenvoudiger. Bij het roteren rond het centerpunt is het best aangeraden om, zoals afgebeeld op de figuur, bij de wielmodules volgens de correcte oriëntatie te werken. Dit zorgt er voor dat het koppel zo optimaal mogelijk gemeten wordt.



Figuur 7.4: Gebruiksaanwijzing ziekenhuisrobot iteratie 2

7.3 Toevoegen LiDAR

Om de performantie van het platform te verhogen, is er voor gekozen bijkomend een LiDAR te implementeren. Dit laat toe muur- en personendetectie te ontwikkelen om nauwkeurige manoeuvres aan een werkstation te vereenvoudigen. De LiDAR gebruikt in de test fase is een Livox MID-360 (figuur 7.5), een hoog performante en toch relatief goedkope 3D LiDAR. Enkele voorname eigenschappen staan beschreven in tabel 7.1.

Tabel 7.1: Livox MID-360 specificaties [45]

Detectie range	40m @ 10% reflectiviteit van het oppervlak 70m @ 80% reflectieviteit van oppervlakken
Field of view	Horizontaal: 360° Verticaal: -7° tot 52°
Framerate	10 Hz
Data-uitwisseling protocol	Ethernet

In de toekomst wordt deze LiDAR vervangen door de twee veiligheidslaserscanners.



Figuur 7.5: Livox MID-360 3D LiDAR [45]

7.3.1 Ingebruiksname

Om vertrouwd te raken met de hardware, zijn enkele initiële testen uitgevoerd om te kijken welke limitaties aan de LiDAR verbonden zijn. Deze observaties zijn in de volgende alinea's beschreven.

Test met Windows 11

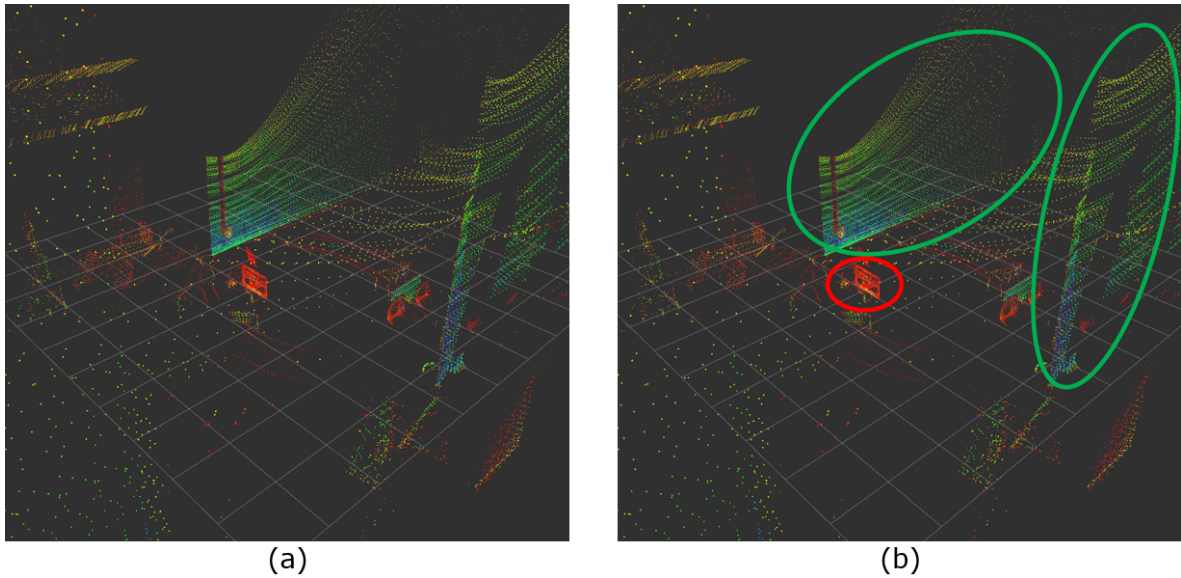
Om snel resultaten te zien, wordt gebruikt gemaakt van Livox Viewer 2 [46]. Deze software geeft direct de puntenwolk die de LiDAR genereert weer in een grafische interface. Belangrijk voor deze test is het IP adres van de RJ45-poort van de computer, waar de LiDAR op aangesloten is, statisch te configureren zoals beschreven in de handleiding van de MID-360.

Test met Ubuntu

Dezelfde software is beschikbaar voor Ubuntu. Echter door de limieten van de hardware waar mee gewerkt wordt voor deze test, is het onmogelijk om deze te gebruiken. Om toch de puntenwolk te visualiseren, wordt RViz en het ROS2-framework gebruikt. Dit heeft als bonus hier een eerste ervaring mee te hebben aangezien deze LiDAR mee in het bestaande systeem geïntegreerd wordt.

Om de MID-360 te gebruiken met ROS2, dient een driver geïnstalleerd te worden. Voor deze driver dient ook een Software Development Kit (SDK) geïnstalleerd te worden. Handleidingen en bestanden hiervoor zijn beschikbaar op de GitHub van Livox [47]. Een aandachtspunt is hier ook weer zorgen dat de IP-adressen, zowel van de poort van de computer als de LiDAR zelf, correct geconfigureerd worden. Indien de bestanden van GitHub gehaald worden, worden ook kleine testscripts mee geïnstalleerd. Om de correcte werking van de SDK te controleren, is er gebruik gemaakt van het quick-start-script en het logger-script. Daarna is de werking van de ROS2-driver gecontroleerd met een geïntegreerd testscript dat de meetpunten in RViz visualiseert. Schermopnames van deze test zijn weergegeven in figuur 7.6.

Deze test is uitgevoerd in de centrale gang van het technologiecentrum in Diepenbeek, waar ACRO gevestigd is. Deelfiguur a representeert de data die de sensor aanlevert. Om te duiden wat in puntenwolk zichtbaar is, zijn er op deelfiguur b enkele ellipsen aangebracht. De rode punten in de rode ellips stelt een standaard monitor voor. De punten in de groene ellipsen stellen de hoge muren van de centrale gang voor.



Figuur 7.6: Schermopname van puntenwolken; (a) basis, (b) met aanduidingen

7.3.2 Detectie van muren

De detectie van de muren verloopt door de puntenwolk om te vormen van het PointCloud2-formaat naar het PCL-pointcloud-formaat van de PCL-library. De PCL-library laat toe het RANSAC-algoritme te gebruiken om vlakken te detecteren. Indien er een vlak gedetecteerd is, wordt de normaalvector van dit vlak berekend alsook de kortste afstand tussen dit vlak en de sensor. Met behulp van de oriëntatie van de sensor wordt ook de hoek tussen het platform en de muur berekend. Op basis van deze hoek zal er beslist worden of het platform zich moet uitlijnen met de muur om een koppeling aan een werkstation toe te laten, of net de muur moet ontwijken. Er wordt enkel naar muren gekeken die zich voor het mobiel platform bevinden, aangezien deze het meest relevante zijn voor het scenario waarin er aan een werkstation gekoppeld moet worden.

Door tijdslimiet zijn hier geen uitgebreide testen mee uitgevoerd. De initiële testen toonden gemixte resultaten wat verdere ontwikkeling van deze functie essentieel maakt.

7.4 Testen en demonstratie van het platformgedrag

Ter validatie van het aangepaste aansturingsalgoritme werden een aantal praktische testen uitgevoerd op de ziekenhuisrobot. Aangezien de hardwarecomponenten voor het uiteindelijke platform met vier wielmodules bij het opstellen van deze thesis nog niet beschikbaar zijn, blijft de werking op dat platform voorlopig beperkt tot een theoretische benadering. Er wordt echter verwacht dat de beschreven logica en gedragspatronen zich analoog zullen gedragen, vermits de software op dynamische wijze schaalbaar is met het aantal aangedreven wielmodules.

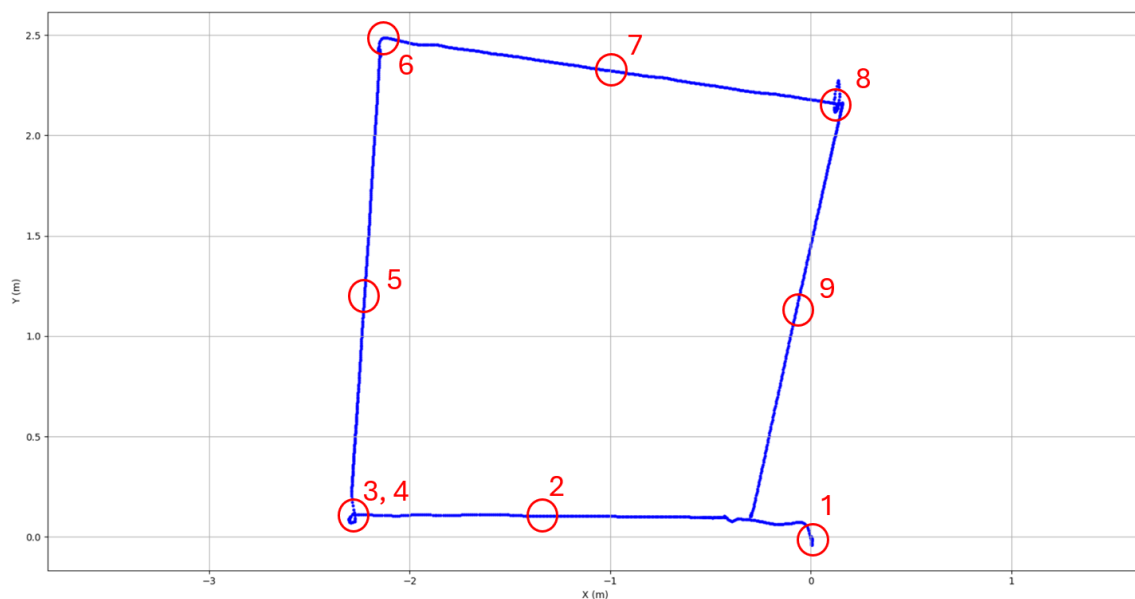
Er werden twee testscenario's opgezet met als doel het gedrag van het platform in typische interactieomstandigheden na te bootsen. In een eerste scenario legde het platform een vierkant pad af, bedoeld om gecontroleerde bewegingen in alle vier richtingen te demonstreren (voorwaarts, achterwaarts, links en rechts). In het tweede scenario werd een draaiend pad gevolgd waarin zowel bochten (links en rechts) als rotatie rond het centrale punt geactiveerd werden. Voor deze testen met de ziekenhuisrobot zijn de detectiegrenzen voor de koppels ingesteld op -4 Nm en 4 Nm.

Bij beide scenario's werd achteraf de odometrie gebruikt om de afgelegde trajecten te reconstrueren. Op basis van deze gegevens werd een visuele representatie opgebouwd van het pad dat het platform aflegde. Het bepalen van een correcte kaart op basis van odometrie is niet evident, zeker met de complexiteit van de omnidirectionele KELO-drives.

7.4.1 Herintegratie van twee wielmodules

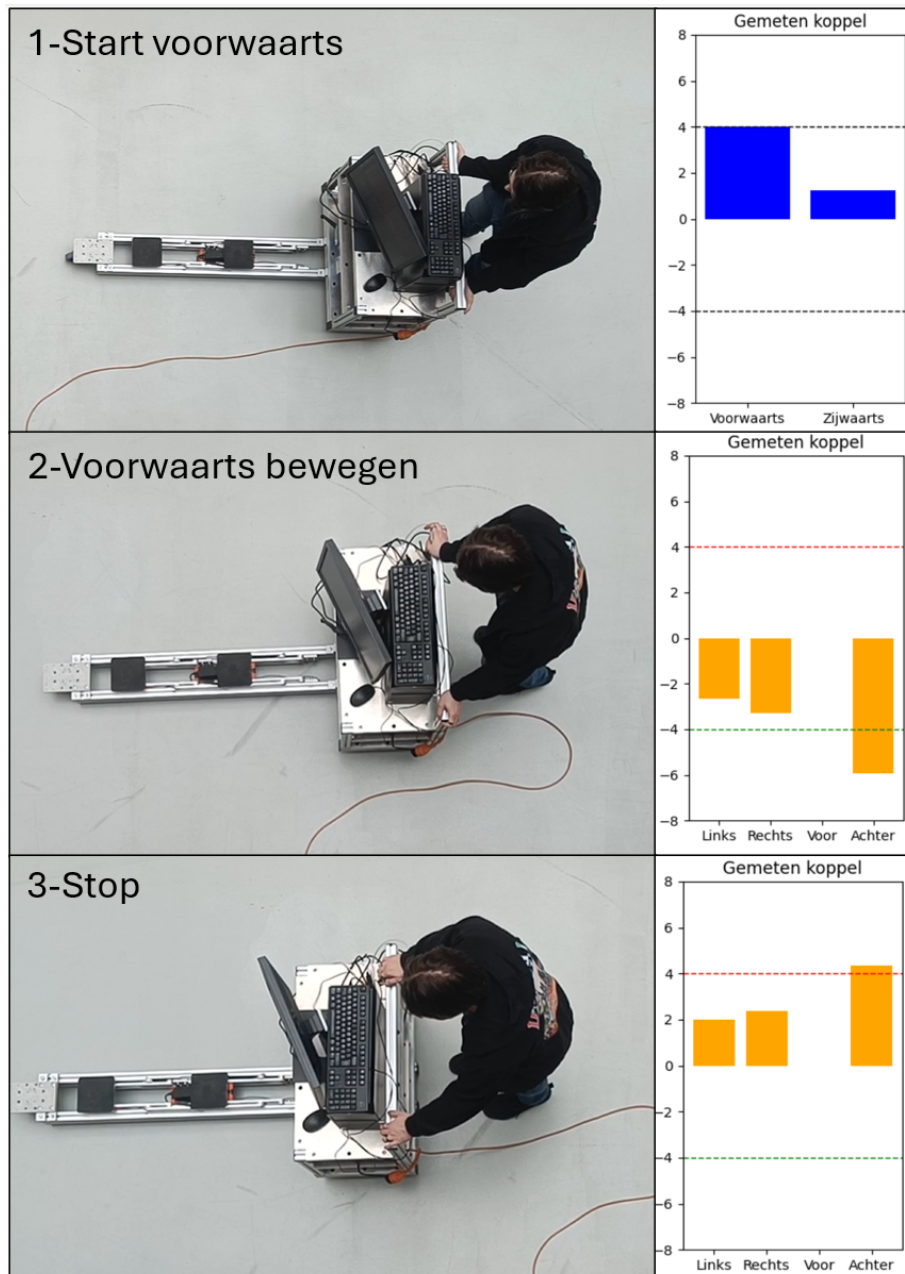
Vierkant parcours

De eerste test demonstreert de correcte detectie en uitvoering van bewegingen in vier richtingen. Hoewel het platform in de werkelijkheid het pad volledig aflegde en terugkeerde naar zijn beginpositie, vertoont de gegenereerde kaart afwijkingen door foutaccumulatie binnen de odometrieberekening. Dit is te wijten aan de complexiteit van de kinematica van KELO-drives. De berekeningen voor de odometrie en het plotten van de kaart moesten ook nog uitgezocht worden, aangezien er geen bestaande formules voor deze kinematica gevonden werden. In de oorspronkelijke KELO robile code was er wel een berekening voor deze odometrie en het plotten, maar onnauwkeurige verwerking van v_y en v_ω leverde slechte resultaten. In bijlage B is het python-script gegeven dat gebruikt was om deze kaart op te stellen. Dit script neemt veel over van de oorspronkelijke aanpak met nog de toevoeging van de correcte verwerking van v_y en v_ω . Figuur 7.7 toont het pad dat het platform aflegde tijdens deze test, met markering van negen relevante situaties waarin het platform zich kan bevinden.



Figuur 7.7: Odometrisch gereconstrueerd pad voor het vierkant parcours

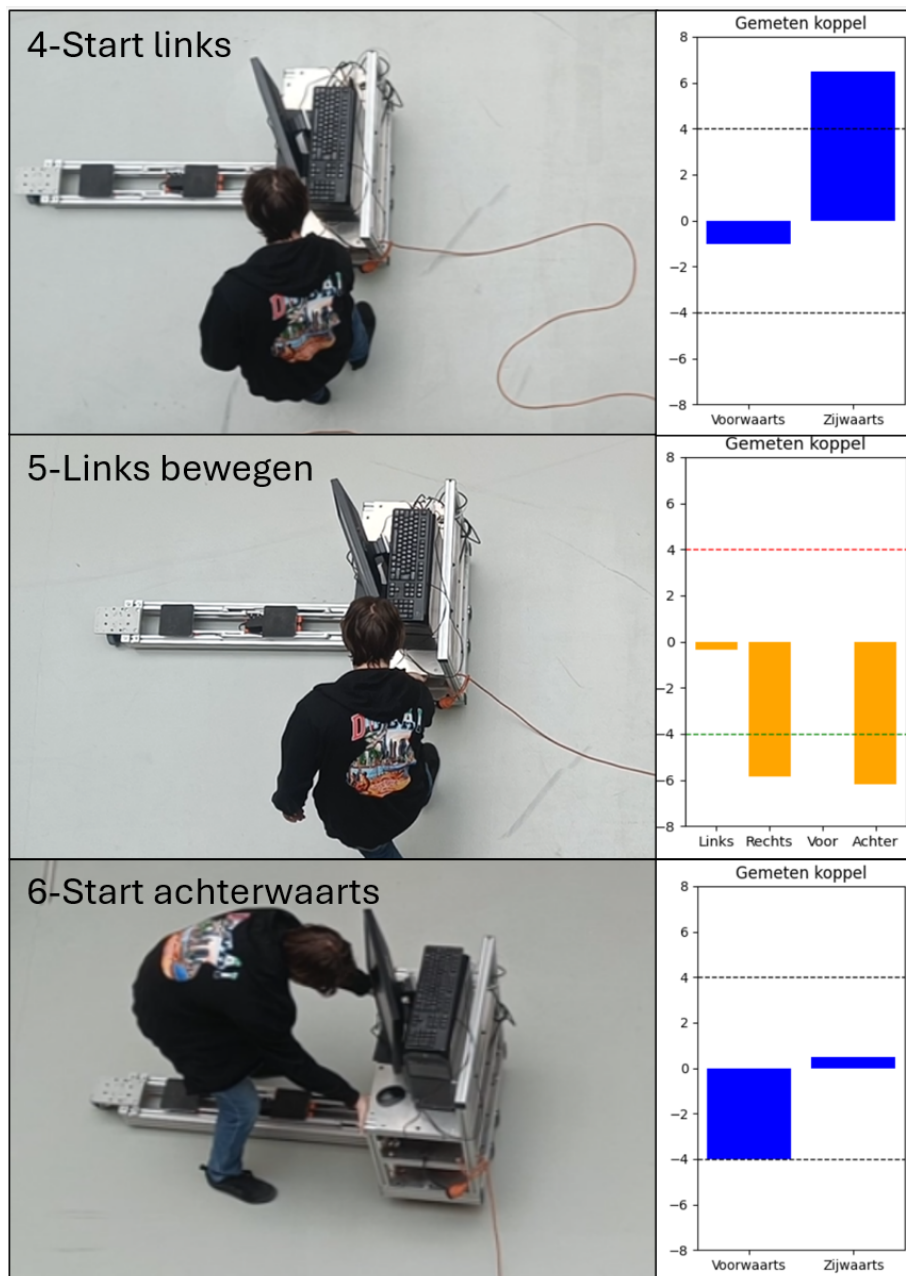
De punten 1, 2 en 3 uit de kaart in Figuur 7.7 worden verder uitgewerkt in Figuur 7.8. Hier wordt per situatie weergegeven welke actie uitgevoerd werd en welke input de operator levert. Rechts van elke afbeelding staan de koppelwaarden afgebeeld die door het platform geïnterpreteerd werden, en die geleid hebben tot de corresponderende actie.



Figuur 7.8: Geregistreerde koppels tijdens situaties 1–3 (voorwaartse beweging en stop)

In situatie 1 oefent de operator een voorwaartse kracht uit op het platform. Dit resulteert in een voorwaarts koppel dat de detectiegrens van 4 Nm overschrijdt, waardoor de voorwaartse modus geactiveerd wordt. Na activatie wordt er niet langer naar het voorwaarts koppel gekeken, maar naar de koppels per zijde. In situatie 2 toont de meting een achterwaarts koppel -4 Nm aan de achterzijde, wat overeenkomt met de versnellingsfase in voorwaartse richting. Bij situatie 3 wordt het platform gestopt en wordt een achterwaarts koppel (> 4 Nm) gedetecteerd. Dit wordt als tegenkracht geïnterpreteerd en beëindigt de beweging.

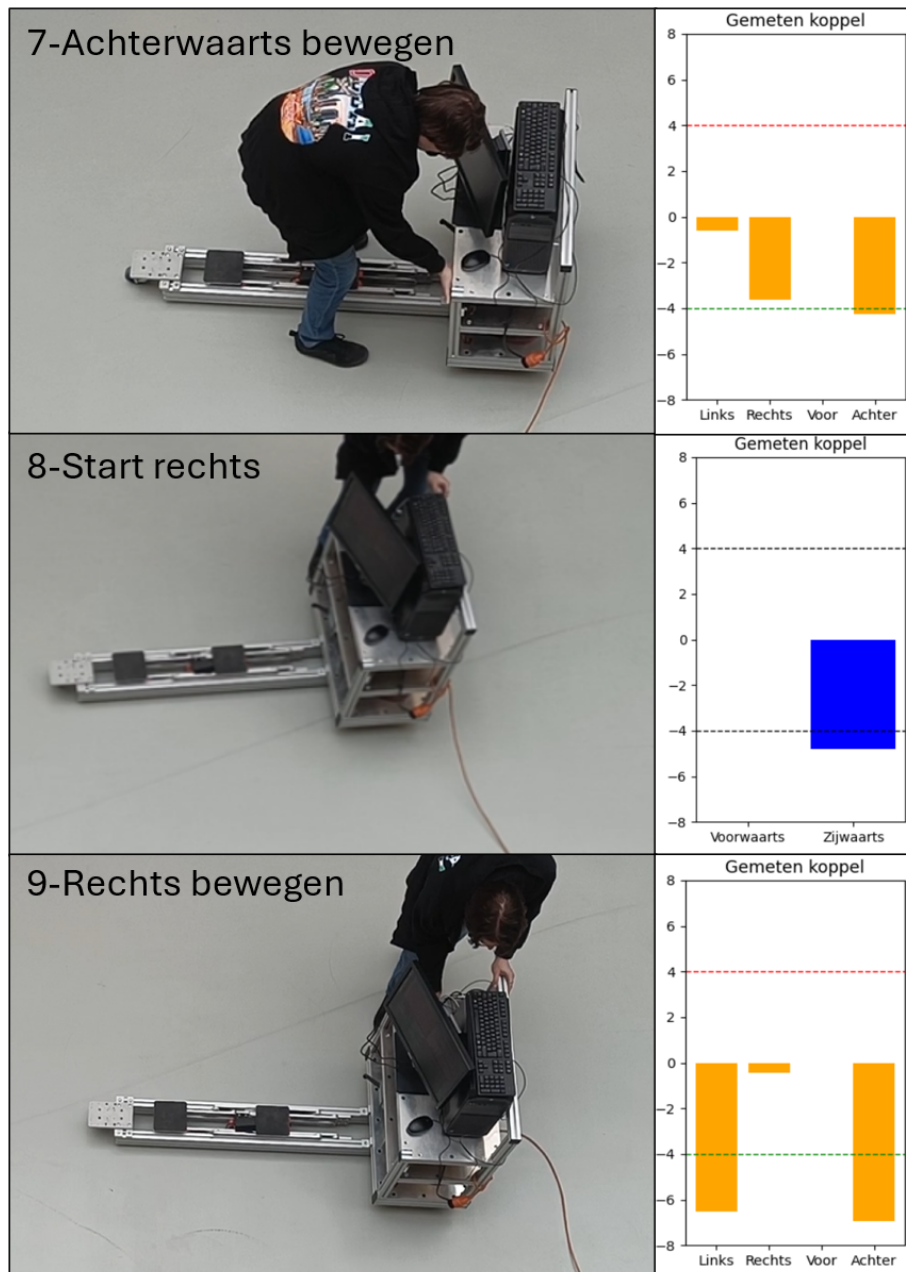
De volgende drie situaties (4, 5 en 6) zijn samengevat in Figuur 7.9. Deze tonen het gedrag bij zijwaartse en achterwaartse interacties. Ook hier zijn telkens de input van de operator en de resulterende koppelmetingen naast elkaar gezet.



Figuur 7.9: Situaties 4–6: detectie en reactie bij zijwaartse en achterwaartse interactie

Bij situatie 4 wordt er een rechtse kracht uitgeoefend, waarbij het zijwaarts koppel de drempel van 4 Nm overschrijdt. Hierdoor wordt de linkse bewegingsmodus geactiveerd. In situatie 5 bevindt het platform zich in beweging; het achterwaarts koppel is negatief (< -4 Nm) en het koppel aan de rechterzijde is groter dan dat aan de linkerzijde, wat bevestigt dat de kracht langs rechts wordt uitgeoefend. In situatie 6 activeert een negatieve waarde van het voorwaarts koppel (< -4 Nm) de achterwaartse beweging.

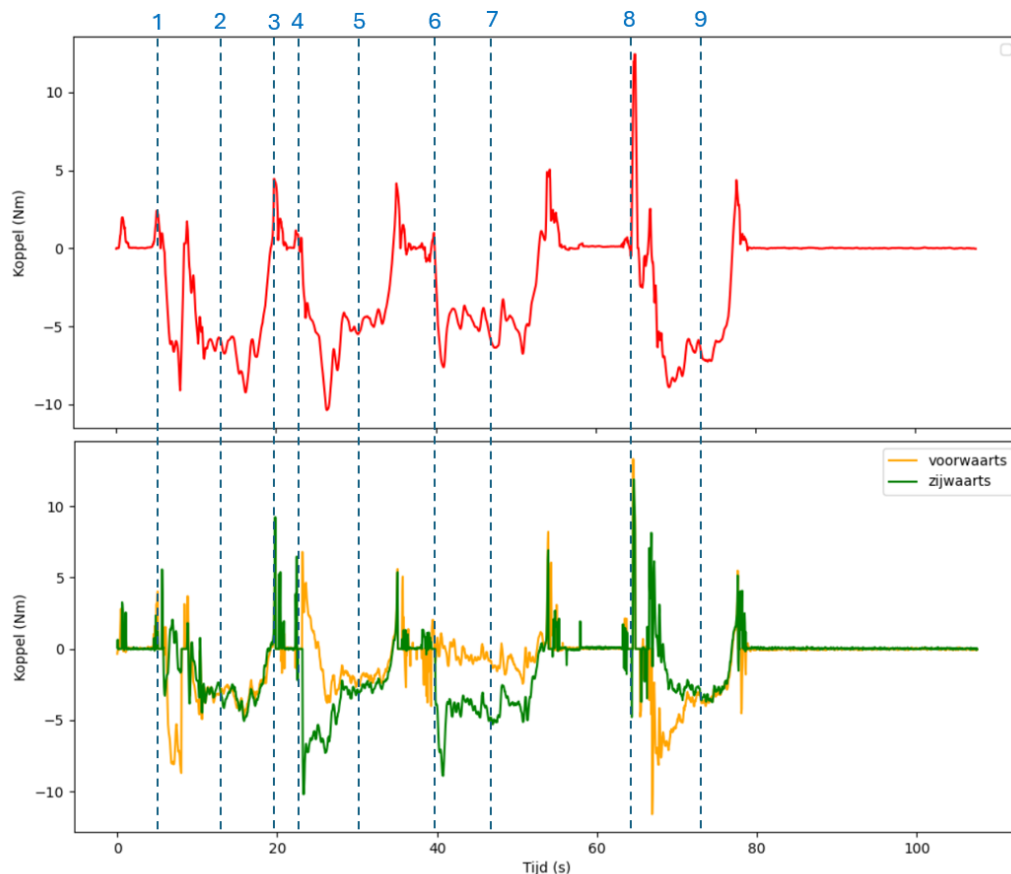
De situaties 7, 8 en 9 zijn weergegeven in Figuur 7.10. Deze illustreren het voortzetten van de achterwaartse beweging en de daaropvolgende interactie langs de linkerzijde.



Figuur 7.10: Situaties 7–9: verdere achterwaartse en linkerzijde-interactie

In situatie 7 blijft het achterwaarts koppel lager dan -4 Nm, wat bevestigt dat de beweging wordt voortgezet. In situatie 8 wordt een zijwaartse kracht naar links uitgeoefend, waardoor het zijwaarts koppel onder -4 Nm daalt en de rechtse modus geactiveerd wordt. In situatie 9 is het achterwaarts koppel opnieuw negatief, en blijkt het koppel aan de linkerzijde groter te zijn dan dat aan de rechterzijde. Dit ondersteunt de interpretatie van een actieve beweging naar rechts.

Tot slot is het ook mogelijk om het volledige verloop van de koppels te plotten en analyseren. Deze plots zijn weergegeven in figuur 7.11. De bovenste plot is het berekende koppel aan de achterzijde, omdat het hier nog steeds over een platform gaat met twee wielmodules is enkel dit koppel relevant. De onderste plot zijn het voorwaartse en zijwaartse koppel. De verschillende situaties zijn met de blauw-gestreepte lijnen aangeduid.

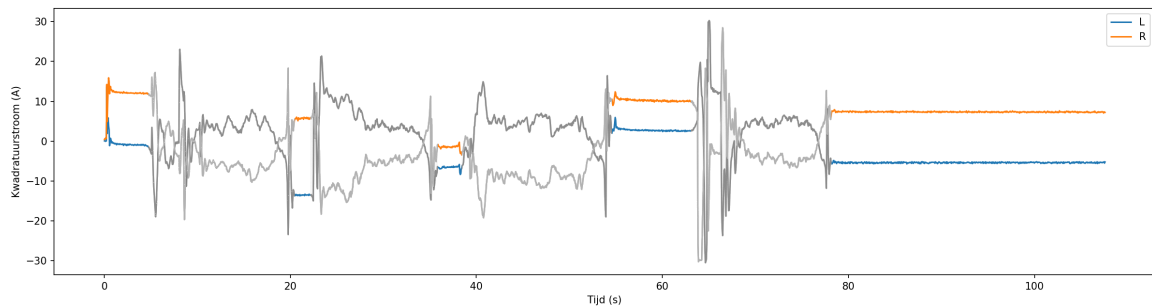


Figuur 7.11: Grafieken van het voorwaarts, zijwaarts en achter koppel voor het vierkant parcours

Op beide grafieken zijn telkens de verschillende fases duidelijk zichtbaar. Elk dal geeft telkens de beweging aan. Deze dalen komen overeen met de situaties 2, 5, 7 en 9. Voor het dal is er de trigger waardoor het platform in beweging gaat, deze zijn het beste zichtbaar op de plot van het voorwaarts en zijwaarts koppel. Bij situaties 1 en 4 is dit een positieve piek en bij situaties 6 en 8 is dit een negatieve piek. De piek van situatie 8 geeft echter eerst iets anders aan. Bij het aansturen naar rechts was hier eerst een ander koppel gemeten, het is echter moeilijk te zeggen of dit rotatie rond center punt geactiveerd had of zijwaartse beweging. Direct na het starten is een tegenkoppel gemeten van > 12 Nm waardoor de beweging onmiddellijk gestopt was. Deze nieuwe grens is een extra veiligheid omdat standaard de trigger van de dead-stop ingesteld is op 4 Nm, maar dit kan pas activeren zodra de beweging 0,5 seconden bezig is. Deze wachttijd is ingesteld omdat bij het oriënteren van de wielmodules bij start koppels buiten de detectiegrenzen kunnen optreden en voor foutpositieven kunnen zorgen. Met deze reden is een extra grens op het viervoud van de normale grens ingesteld zonder deze wachttijd, voor deze eerste 0,5 seconden toch veilig te kunnen laten verlopen.

Bovenstaande grafieken ondersteunen verder de argumentatie bij algoritme 7.5.2 in deel 7.2.3. Hier is vermeld dat er telkens gekozen wordt voor het koppel van de zijdes waar er zich twee wielmodules bevinden. Het koppel van de linker en rechter zijde zijn hier wel niet geplot maar volgen nog steeds gelijke trend met het voorwaarts en zijwaarts koppel. In de opstelling met twee wielmodules is het enige verschil het voortschrijdend gemiddelde, dus zijn de vormen van de grafieken gelijkaardig. Het voornaamste voordeel zijn de pieken bij de start van de zijwaartse beweging in situaties 5 en 9 die weg gewerkt worden.

Om een beter beeld te krijgen van de verwerking van de inkomende kwadratuurstroom naar het bijhorende koppel, wordt dit geplot in figuur 7.12. De kwadratuurstroom is afkomstig uit de wielmodule die het zijwaarts koppel meet. De momenten dat het platform stil staat zijn in kleur en de momenten dat het platform in beweging is zijn zwart wit.



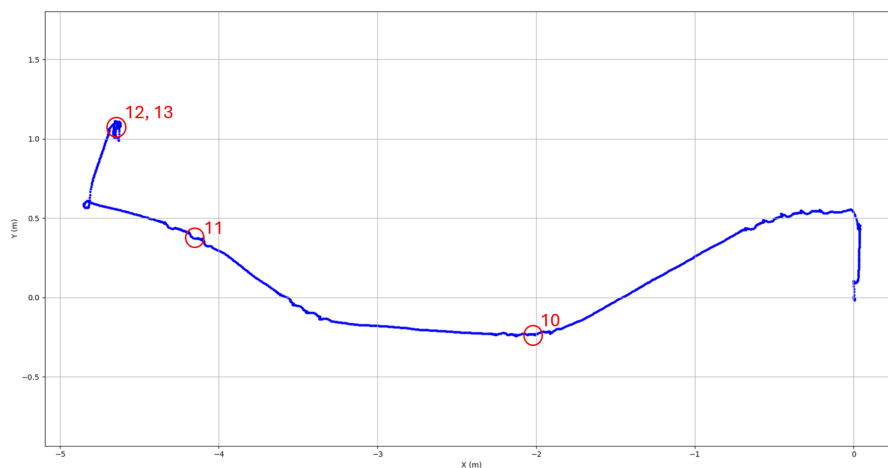
Figuur 7.12: Grafiek van de kwadratuurstroom voor het zijwaarts koppel van het vierkant parcours

Opvallend aan de kwadratuurstroom is dat deze, na het uitvoeren van een beweging, telkens een andere waarde aan neemt bij rust. Toch wordt deze onregelmatigheid goed verwerkt en zijn de waarden voor de koppels bij stilstand rond nul.

Draaiparcours

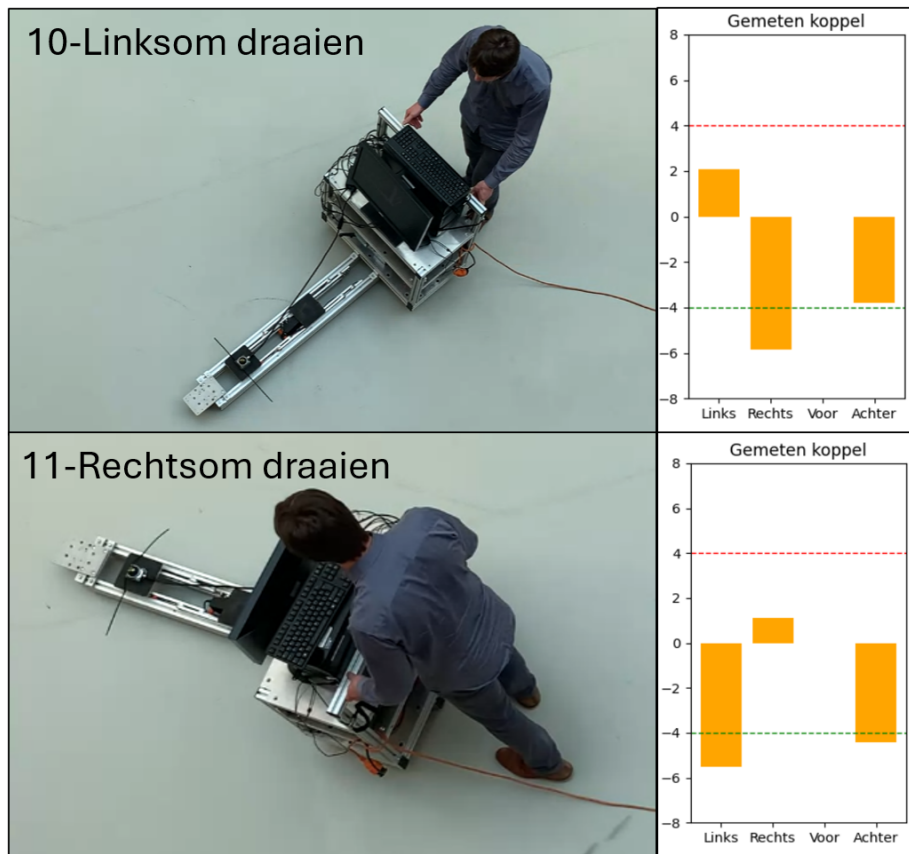
De tweede test focust op het gedrag van het platform tijdens rotatiebewegingen. Hierbij wordt zowel het links en rechts draaien tijdens beweging als het draaien rond het centrale punt van het platform geëvalueerd. Ook hier is het pad opgebouwd via odometrie en vertoont het afwijkingen ten opzichte van het werkelijke pad, om dezelfde redenen als bij het vierkant parcours. De test toont echter duidelijk de correcte interpretatie van de koppels, en dus het functioneel gedrag van de code.

In Figuur 7.13 is het volledige pad weergegeven met markering van vier situaties waarin rotatie- of draaibewegingen geactiveerd of voortgezet werden.



Figuur 7.13: Gereconstrueerd pad voor het draaiparcours

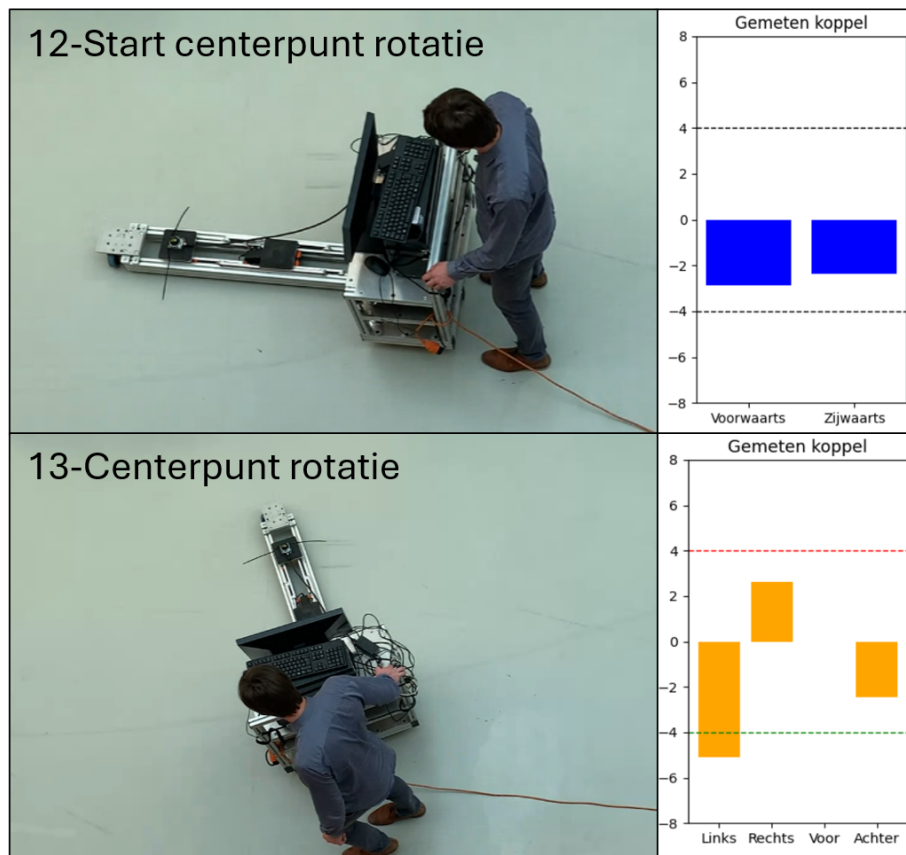
De situaties 10 en 11 worden gevisualiseerd in Figuur 7.14. Elke situatie toont een bochtbeweging tijdens platformverplaatsing, waarbij de draairichting bepaald wordt door de verdeling van het koppel over de linker- en rechterzijde.



Figuur 7.14: Situaties 10 en 11: roteren tijdens beweging naar links en rechts

In situatie 10 wordt er linksom gedraaid. Dit blijkt uit een koppel $< -4Nm$ aan de rechterzijde, terwijl aan de linkerzijde een licht positief koppel wordt gemeten. Het platform interpreteert dit correct als een bocht naar links. In situatie 11 gebeurt het omgekeerde: het linker koppel is $< -4Nm$, en het rechter licht positief. Dit activeert een bocht naar rechts. In beide gevallen toont de koppeldistributie duidelijk de asymmetrie die nodig is om een draaiende beweging correct te starten.

De situaties 12 en 13 zijn weergegeven in Figuur 7.15. Deze tonen de detectie en voortzetting van een centrale rotatiebeweging, onafhankelijk van translatie.



Figuur 7.15: Situaties 12 en 13: interpretatie en voortzetting van rotatie rond het centrum

In situatie 12 is het platform nog in stilstand. Zowel het voorwaartse als het zijwaartse koppel bevinden zich tussen $-2Nm$ en $-4Nm$, wat binnen de ingestelde drempel ligt voor het activeren van een rotatie rond het centerpunt. In situatie 13 bevindt het platform zich effectief in rotatie. Zoals beschreven in 7.2.3, wordt bij voortgezette centrale rotatie gekeken naar de zijde met het laagste koppel. In dit geval is dit de linkerzijde, waar een koppel $< -4Nm$ gemeten wordt. Dit bevestigt dat de juiste voortzettingslogica toegepast wordt.

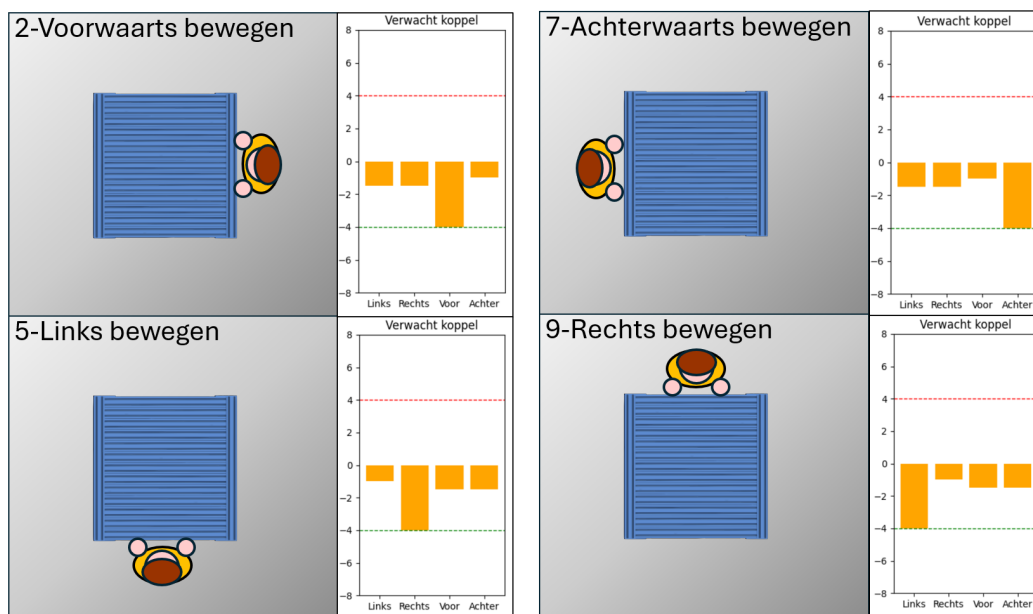
7.4.2 Theoretische werking van vier wielmodules

Hoewel het vierwielige platform nog niet fysiek beschikbaar was tijdens de testfase, laat de structuur van de software toe om de verwachte werking theoretisch te benaderen. Op basis van de bestaande groeperings- en koppelberekeningslogica wordt verondersteld dat het globale gedrag grotendeels analoog zal zijn aan dat van de ziekenhuisrobot.

Wel wordt verwacht dat de detectiegrenzen licht aangepast moeten worden. Door de toegenomen resolutie in krachtmeting en de verdeling van de belasting over meerdere wielmodules, zullen de resulterende koppels per zijde hoger liggen dan in de tweewielige configuratie. Er wordt echter niet verondersteld dat de operator daardoor significant harder moet duwen om een reactie uit te lokken. De aangepaste detectiegrenzen compenseren de spreiding van de meetwaarden, wat toelaat om bij gelijkaardige externe kracht toch een correcte activatie van beweging te verkrijgen.

Voor de meeste situaties wordt hetzelfde gedrag verwacht als bij het platform met twee wielmodules. Vier specifieke gevallen gaan mogelijk afwijkend gedrag vertonen: situaties 2, 5, 7 en 9. Deze gevallen hebben betrekking op de koppels in een bepaalde richting terwijl het platform al in beweging is. Waar bij het platform met twee wielmodules enkel de achterzijde in rekening gebracht kan worden, gaat bij de vier wielmodules elke zijde in rekening gebracht kunnen worden.

Figuur 7.16 illustreert deze vier situaties. Per situatie wordt een schets getoond van de operator die kracht uitoefent op het platform, samen met de bijhorende plots van de vier koppelwaarden per zijde.



Figuur 7.16: Verwachte koppels bij situaties 2, 5, 7 en 9 voor het mobiel platform met vier wielmodules

In elk van de bovenstaande situaties is te zien dat het verwachte koppel op de zijde waar de kracht wordt uitgeoefend, het grootst is. In toekomstig werk zou dit bevestigd moeten worden of dit in de realiteit ook effectief bekomen wordt.

7.5 Conclusie

Hoewel het binnen dit werk niet mogelijk was om de tweede iteratie van de aansturing volledig te testen op het platform met vier wielmodules, en de LiDAR-functionaliteit wegens tijdsgebrek niet operationeel werd, werd een robuust resultaat bekomen op het platform met twee wielmodules. De tweede iteratie van de code vormt een duidelijke verbetering ten opzichte van de oorspronkelijke versie: het platform is nu logisch en consistent bestuurbaar, met een directe vertaling van externe krachten naar beweging.

De nieuwe architectuur maakt het bovendien mogelijk om flexibel te schakelen tussen configuraties met twee, drie of vier wielmodules. De groepering van modules per zijde en oriëntatie, de dynamische berekening van koppels en de automatische selectie van de dominante zijde zorgen voor een schaalbare en uitbreidbare structuur. Ook de kalibratieprocedure en parametrisatie dragen bij tot een efficiëntere afstelling.

Uit de praktijktesten blijkt dat de interpretatie van interacties stabiel verloopt, en dat het platform betrouwbaar reageert op variërende externe krachten. Dit vormt een stevige basis voor verdere uitbreiding, waaronder de fysieke integratie van de vier wielmodules, optimalisatie van detectiedrempels, en activering van geavanceerde functionaliteiten zoals LiDAR-gebaseerde muurvolgning.

De opgestelde code is zo opgebouwd dat toekomstige uitbreidingen en validaties vlot kunnen worden geïntegreerd. Ondanks technische en praktische beperkingen is er doordacht en doelgericht gewerkt aan een code die makkelijk toepasbaar is voor toekomstige mobiele platformen.

Hoofdstuk 8

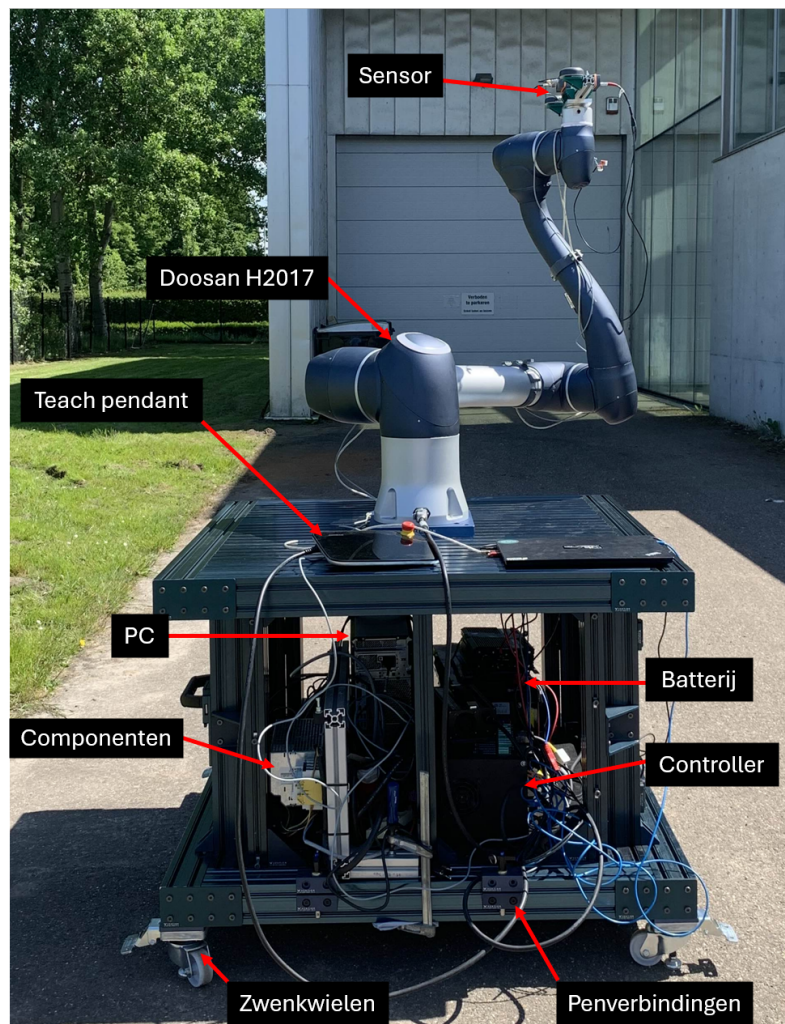
Use case

Gelijktijdig met deze masterproef liep een andere masterproef die gebruik diende te maken van het mobiel platform. Deze werd uitgevoerd door Jonah Fastré met als interne promotor Eric Demeester, interne begeleider Anton Melnikov en externe promotor Marnix Volckaert van het bedrijf Septentrio.

Die masterproef had als doel het kalibreren van een sensor, om dit te vergemakkelijken wordt deze op de Doosan H2017 gemonteerd. Deze kalibratie dient buiten plaats te vinden dus moet het mogelijk zijn om de cobot naar daar te verplaatsen. Dit brengt al een eerste use case mee om het platform te testen en te controleren of het voldoet aan de verwachtingen.

8.1 Opstelling

De testen van deze use case vonden al plaats vooraleer het elektrisch gedeelte, volgens het ontwerp in deze thesis, klaar was en gemonteerd kon worden. De noodzakelijke componenten zijn daarom hier niet ordelijk bevestigd maar provisorisch geplaatst om toch de nodige functionaliteit te hebben. In figuur 8.1 is de opstelling voor de use case weergegeven.



Figuur 8.1: Opstelling use case met Doosan H2017 en bijhorende componenten

Centraal op het platform staat de Doosan H2017 gemonteerd met op de eindeffector de nodige sensor. In het rechtse compartiment bevinden zich de batterij, oplader van de batterij, controller van de Doosan en de omvormer. Het linkse compartiment bevat de PC die in deze opstelling niet gebruikt werd. Op de grijze profielen in dit deel zitten de nodige veiligheidscomponenten voor de aankoppeling op netstroom en componenten voor de communicatie. Onder het platform zijn de zwenkwielen met uitklapbare poten gemonteerd.

De panelen aan de zijkanten, gemaakt voor het afschermen van de interne componenten, staan niet gemonteerd in de opstelling. Voornamelijk omdat het een tijdelijke opstelling is en de componenten uitsteken, maar ook omdat er nog geen openingen in het bovenblad zijn voorzien om kabels langs te begeleiden.

8.2 Bevindingen

Uit het eerste gebruik van het platform volgden al enkele opmerkingen en feedback over de verplaatsing en integriteit. Tijdens verplaatsing over ongelijke ondergrond is ondervonden dat doordat het onderstel geen massieve plaat is maar uit meerdere extrusieprofielen bestaat, het een minimale vorm van flexibiliteit toelaat en het mogelijk is voor de vier wielen om ten alle tijde contact te hebben met de ondergrond. Daarnaast is bij de poort om naar buiten te gaan een verhoging van enkele centimeters waar het platform deels over getild moest worden. Het wisselen van de zwenkwielen met de KELO 100 drives zou dit manouvre moeten vergemakkelijken, dankzij het toevoegen van de aandrijving met krachtgedreven assistentie, om dit obstakel beter te overbruggen zonder dat extra fysieke arbeid nodig is.

Op het vlak van integriteit is het platform algemeen voldoende stabiel, zeker voor het eigengewicht van de arm en componenten te dragen. Wanneer de Doosan echter kort bij singulariteiten zat en plotse bewegingen aan hoge snelheden uitvoerde waren er kleine schokken hoorbaar van het platform. Mogelijke verplaatsing of loskomen van de grond is echter niet onmiddellijk waargenomen. Vermoedelijk waren de hoorbare schokken te wijten aan de extrusie profielen die niet perfect tegen elkaar aan staan. Indien het één vaste lasconstructie zou zijn zal dit niet optreden.

Uiteindelijk is het ontworpen platform stabiel genoeg, met voldoende ruimte voor de nodige componenten voor het uitvoeren van de andere gelijklopende masterproef.

Hoofdstuk 9

Besluit en vooruitblik

In deze thesis werd een mobiel platform ontwikkeld als basis voor een Doosan H2017-cobotarm. Het project omvatte het mechanisch ontwerp, een stabiliteitsanalyse, een elektrisch ontwerp en de ontwikkeling van krachtgedreven besturingsalgoritmes. De mechanische structuur werd succesvol ontworpen en opgebouwd, waarbij gebruik werd gemaakt van een robuust frame uit extrusieprofielen.

Een gedetailleerde stabiliteitsanalyse bevestigde dat het platform in staat is de dynamische belastingen van de cobotarm te weerstaan. De totale massatraagheid van de arm, inclusief maximale payload op maximale reikwijdte, werd berekend op $I_{\text{cob}} = 98,79 \text{ kg} \cdot \text{m}^2$, met een verwachte maximale hoekversnelling van $\alpha = 0,64 \text{ rad/s}^2$ oftewel $36,67^\circ/\text{s}^2$. Voor het behouden van stabiliteit bij deze belastingen bleek een massa van minstens $m_{\text{mp}} = 83,22 \text{ kg}$ nodig te zijn voor het frame, terwijl het effectieve gewicht volgens de Vention MachineBuilder 181,30 kg bedraagt. Het platform is dus met marge stabiel tijdens operationeel gebruik.

Uit de analyse blijkt wel dat één specifieke wielmodule (onder de cobotarm) in extreme configuraties overbelast raakt deze module zou tot 161,81 kg moeten dragen, wat boven het toegelaten maximum van 125 kg ligt. Dit probleem kan vermeden worden door bij gebruik van de cobot extra stabilisatoren te voorzien. Belangrijker is echter dat tijdens verplaatsing tussen werkstations geen enkele wielmodule overbelast wordt, op voorwaarde dat de arm in een vooraf bepaalde transportpositie geplaatst wordt die de belasting herverdeelt.

Parallel aan het mechanisch ontwerp werd ook het elektrisch ontwerp opgesteld. Hoewel de implementatie hiervan niet gerealiseerd kon worden binnen de tijdspanne van deze thesis. Dit komt ook door levertermijnen en complicaties met de wielmodules. Dit ontwerp biedt desondanks een solide basis voor verdere ontwikkeling en integratie van de elektrische componenten in toekomstig werk.

Op het vlak van softwareontwikkeling werd een algoritme voor krachtgedreven aansturing in twee iteraties getest op een testplatform met twee wielmodules. De eerste versie bleek functioneel maar moeilijk bestuurbaar, terwijl de tweede iteratie significant vlotter en intuïtiever bestuurbaar was. Dit toont aan dat stapsgewijze verfijning van het algoritme de prestaties van het systeem merkbaar kan verbeteren.

Een uitbreiding op dit tweede algoritme voorzag de integratie van een LiDAR-sensor voor muur-detectie en automatische uitlijning met muren, om zo gecontroleerd evenwijdig langs obstakels te kunnen bewegen. Wegens tijdsgebrek kon deze uitbreiding niet verder worden uitgewerkt, maar dit vormt een waardevolle piste voor toekomstig onderzoek.

Eveneens werd vastgesteld dat de keuze om gebruik te maken van interne sensoren van de wiel-modules voor de detectie van externe krachten het ontwikkelingsproces heeft bemoeilijkt. Een alternatieve aanpak, met bijvoorbeeld externe kracht- of momentsensoren, zou de implementatie wellicht eenvoudiger en nauwkeuriger maken.

Tot slot toonde een eerste praktische use case aan dat het ontworpen platform geschikt is als mobiele basis voor de Doosan H2017-cobotarm, wat de haalbaarheid van het concept aantoont en ruimte biedt voor verdere verfijning en toepassing in industriële contexten.

Toekomstig werk

Op basis van het werk dat in het kader van deze masterproef is uitgevoerd, zijn er enkele logische vervolgstappen voor het project.

Zo is de integratie van de elektrische componenten in het Vention-platform een vereiste om de algoritmes hierop te testen. Aansluitend dienen ook nog gedetailleerdere elektrische schema's opgesteld te worden die het detecteren en oplossen van fouten in de toekomst significant kunnen vergemakkelijken.

Nadien kan de implementatie en validatie van het tweede algoritme op het platform met vier wielmodules gebeuren. Alsook de verdere uitwerking van de LiDAR-gebaseerde muurvolging.

Het onderzoeken van alternatieve methoden voor krachtmeting, met nadruk op externe sensor-technologieën voor verhoogde nauwkeurigheid en eenvoud. Vooral op het vlak van draaien tijdens een voor- of achterwaartse beweging is er nog ruimte voor verbetering. Daarnaast kan er ook nog onderzoek gedaan worden naar automatische uitlijntechnieken op basis van een visiesysteem.

Literatuurlijst

- [1] Sirris, “Sirris,” 2022. Online beschikbaar via: <https://www.sirris.be/nl> (geraadpleegd op 22 oktober 2024).
- [2] KU Leuven, “Automation, computer vision and robotics,” 2024. Online beschikbaar via: <https://iiw.kuleuven.be/onderzoek/acro> (geraadpleegd op 22 oktober 2024).
- [3] Doosan Robotics Inc., “H-series, H2017,” 2024. Online beschikbaar via: <https://www.doosanrobotics.com/en/products/series/h2017> (geraadpleegd op 10 oktober 2024).
- [4] robu.in, “80mm-a mecanum wheel compatible with 6.7mm coupling (pack of 4)-yellow.” Online beschikbaar via: <https://robu.in/product/80mm-a-mecanum-wheel-compatible-with-6-7mm-coupling-pack-of-4-yellow/> (geraadpleegd op 25 november 2024).
- [5] I. B. Erland, “Wheels for a course stable selfpropelling vehicle movable in any desired direction on the ground or some other base.” US3876255A, 8 april 1975, <https://worldwide.espacenet.com/patent/search/family/023182799/publication/US3876255A?q=pn=US3876255> (geraadpleegd op 15 oktober 2024).
- [6] H. Shah and Fresh Consulting, “How to build a swerve-drive robot,” 2023. Online beschikbaar via: <https://www.freshconsulting.com/insights/blog/how-to-build-a-swerve-drive-robot/> (geraadpleegd op 15 Augustus 2024).
- [7] Neobotix GmbH, “Drive module for your omnidirectional robots,” 2025. Online beschikbaar via: <https://www.neobotix-robots.com/products/robot-components/omnidrivemodule> (geraadpleegd op 15 Augustus 2024).
- [8] U. Caliskan, F. U. Rios, W. Decré, and E. Aertbeliën, “Dual constraint-based controllers for wheeled mobile manipulators,” in *2022 IEEE 18th International Conference on Automation Science and Engineering (CASE)*, pp. 1001–1008, 2022.
- [9] J. H. Lee, B. K. Kim, T. Tanikawa, and K. Ohba, “Kinematic analysis on omni-directional mobile robot with double-wheel-type active casters,” in *2007 International Conference on Control, Automation and Systems*, pp. 1217–1221, 2007.
- [10] Y.-J. Ryoo, “Development of an autonomous mobile robot platform for smart farms,” *International Journal of Fuzzy Logic and Intelligent Systems*, 2023.
- [11] M. Jafari, “Get rolling with omni-directional wheels,” *servomagazine*, 2019. MEER INFO NODIG.

- [12] P. Prodoehl, “Omni wheel robot,” 2023. Online beschikbaar via: <https://learn.browndoggadgets.com/Guide/Omni+Wheel+Robot/310> (geraadpleegd op 27 december 2024).
- [13] Neobotix GmbH, *MMO-700*. Neobotix, September 2023. Available at https://www.neobotix-robots.com/fileadmin/images/downloads/Bedienungsanleitung_2022/MMO-700_Englisch_01.pdf.
- [14] KELO Robotics GmbH, “Kelo drive 100.” Online beschikbaar via: <https://shop.kelo-robotics.com/product/kelo-drive-100/> (geraadpleegd op 20 Augustus 2024).
- [15] M. Schwarz, M. Beul, D. Droeschel, T. Klamt, C. Lenz, D. Pavlichenko, and T. Rodehuts-kors, “Drc team nimbro rescue: Perception and control for centaur-like mobile manipulation robot momaro,” *Journal of Field Robotics*, vol. 34, no. 2, pp. 400–425, 2017.
- [16] Swerve Drive Specialties, “Swerve module kits.” Online beschikbaar via: <https://www.swervedrivespecialties.com/collections/kits> (geraadpleegd op 20 Augustus 2024).
- [17] Hunan Yousheng Technology Co. Ltd., “Yousheng agv 48v 400w steerable drive wheel unit with servo motor 48v 400w tzbot.” Online beschikbaar via: https://www.alibaba.com/product-detail/YOUSHENG-AGV-48V-400W-Steerable-Drive_1600488555868.html?module=company (geraadpleegd op 2024).
- [18] Vention Inc., “About vention,” 2025. Online beschikbaar via: <https://vention.io/about> (geraadpleegd op 14 september 2024).
- [19] M. Guertler et al., “When is a robot a cobot? moving beyond manufacturing and arm-based cobot manipulators,” *Proceedings of the Design Society*, vol. 3, pp. 3889–3898, 2023.
- [20] Universal Robots A/S, “UR5,” 2024. Online beschikbaar via: <https://www.universal-robots.com/products/ur5-robot/> (geraadpleegd op 10 oktober 2024).
- [21] KUKA AG, “LBR iiwa,” 2024. Online beschikbaar via: <https://www.kuka.com/en-us/products/robotics-systems/industrial-robots/lbr-iiwa> (geraadpleegd op 10 oktober 2024).
- [22] MENNEKES Elektrotechnik GmbH & Co. KG, “Standard plugs,” 2024. Online beschikbaar via: <https://www.mennekes.org/industry/products/product-specific/plugs/standard-plugs/> (geraadpleegd op 25 november 2024).
- [23] Pneumatic Components Ltd (PCL), “Interchange range,” 2024. Online beschikbaar via: <https://euro.pclairtechnology.com/couplings-adaptors/interchange-range> (geraadpleegd op 25 november 2024).
- [24] EtherCAT Technology Group, “Ethercat product guide,” 2025. Online beschikbaar via: <https://www.ethercat.org/en/products.html> (geraadpleegd op 20 maart 2025).
- [25] “Richtlijn 2006/42/eg van het europees parlement en de raad van 17 mei 2006 betreffende machines en tot wijziging van richtlijn 95/16/eg (herschikking).” Online beschikbaar via: <http://data.europa.eu/eli/dir/2006/42/oj> (geraadpleegd op 12 maart 2025).
- [26] Pilz GmbH & Co. KG, “Pnoz m b1,” 2025. Online beschikbaar via: <https://www.pilz.com/en-INT/eshop/Small-controllers-PNOZmulti/Small-controllers-PNOZmulti-2/>

PN0Zmulti-2-base-units/PN0Zmulti-2-base-units/PN0Z-m-B1/p/772101?etcc_cu=onsite&etcc_cmp_onsite=PN0Zmulti&etcc_med_onsite=selection_guide (geraadpleegd op 14 maart 2025).

- [27] Beckhoff Automation GmbH & Co. KG, “Ek1960 — twinsafe compact controller,” 2025. Online beschikbaar via: <https://www.beckhoff.com/nl-be/products/i-o/ethercat-terminals/elx9xx-twinsafe/ek1960.html> (geraadpleegd op 14 maart 2025).
- [28] SICK AG, “Veiligheidscontrollers: Flexi soft fx3-cpu000000,” 2025. Online beschikbaar via: <https://www.sick.com/be/nl/catalog/producten/safety/veiligheidscontrollers/flexi-soft/fx3-cpu000000/p/p80477?tab=detail> (geraadpleegd op 14 maart 2025).
- [29] SICK AG, “Veiligheidscontrollers: Flexi compact flx3-cpuc100,” 2025. Online beschikbaar via: <https://www.sick.com/be/nl/catalog/producten/safety/veiligheidscontrollers/flexi-compact/flx3-cpuc100/p/p661374?tab=detail> (geraadpleegd op 14 maart 2025).
- [30] Raspberry Pi Ltd, “Revolution pi industrial computers,” 2024. Online beschikbaar via: <https://www.raspberrypi.com/success-stories/revolution-pi-industrial-computers/> (geraadpleegd op 15 november 2024).
- [31] KUNBUS GmbH, “White labeling and customization,” 2024. Online beschikbaar via: <https://revolutionpi.com/en/customization> (geraadpleegd op 25 november 2024).
- [32] OnLogic, “Industrial pc,” 2024. Online beschikbaar via: <https://www.onlogic.com/store/computers/industrial/> (geraadpleegd op 15 november 2024).
- [33] Canonical Ltd., “The story of ubuntu,” 2024. Online beschikbaar via: <https://ubuntu.com/about> (geraadpleegd op 25 november 2024).
- [34] Canonical Ltd., “What is ros,” 2024. Online beschikbaar via: <https://ubuntu.com/robotics/what-is-ros> (geraadpleegd op 30 november 2024).
- [35] Open Robotics, “Ros - robot operating system,” 2024. Online beschikbaar via: <https://www.ros.org/> (geraadpleegd op 30 november 2024).
- [36] Opsinox NV, “Home,” 2024. Online beschikbaar via: <https://www.opsinox.com/nl/home> (geraadpleegd op 20 december 2024).
- [37] H. Austerlitz, *Data Acquisition Techniques Using PCs (Second Edition)*. Academic Press, 2003.
- [38] Advanced Navigation Pty Ltd, “Inertial measurement unit (imu) – an introduction,” 2023. Online beschikbaar via: <https://www.advancednavigation.com/tech-articles/inertial-measurement-unit-imu-an-introduction/> (geraadpleegd op 04 juni 2025).
- [39] M. Cramer and E. Demeester, “Dynamic load analysis of a dual-arm mobile manipulator performing a lateral lifting operation,” 2024.
- [40] T. Floré and PILZ, “Basiscursus functionele veiligheid versie 1.0,” 2024.
- [41] SICK AG, “Veiligheidsschakelaar re2,” 2025. Online beschikbaar via: <https://www.sick.com/be/nl/catalog/producten/safety/veiligheidsschakelaar/re2/> (geraadpleegd op 14 maart 2025).

- [42] SICK AG, “Veiligheidslaserscanner: microscan3 mics3-ccaz55an1p01,” 2025. Online beschikbaar via: <https://www.sick.com/be/nl/catalog/producten/safety/veiligheidslaserscanner/microscan3/mics3-ccaz55an1p01/p/p678415?tab=detail> (geraadpleegd op 14 maart 2025).
- [43] Beckhoff Automation GmbH & Co. KG, “Beckhoff information system,” 2025. Online beschikbaar via: <https://infosys.beckhoff.com/english.php?content=../content/1033/ethercatsystem/2469078667.html&id=> (geraadpleegd op 29 maart 2025).
- [44] D. De Schepper, I. Dekker, A. Melnikov, G. Schouterden, K. Kellens, and E. Demeester, “Development of a mobile platform for the transportation of medical carts in hospital environments,” 2024.
- [45] Livox Technology Company Limited Co., Ltd., “Mid-360,” 2025. Online beschikbaar via: <https://www.livoxtech.com/mid-360> (geraadpleegd op 10 mei 2025).
- [46] Livox Technology Company Limited Co., Ltd., “Downloads,” 2025. Online beschikbaar via: <https://www.livoxtech.com/downloads> (geraadpleegd op 15 mei 2025).
- [47] Livox Technology Company Limited Co., Ltd., 2025. Online beschikbaar via: <https://github.com/Livox-SDK> (geraadpleegd op 15 mei 2025).

Bijlage A

**Bijlage - Parameters voor berekening
kinematica en dynamica Doosan H2017**

Parameter for Calculations of kinematic and dynamics

Document Version: V1.0



DOOSAN

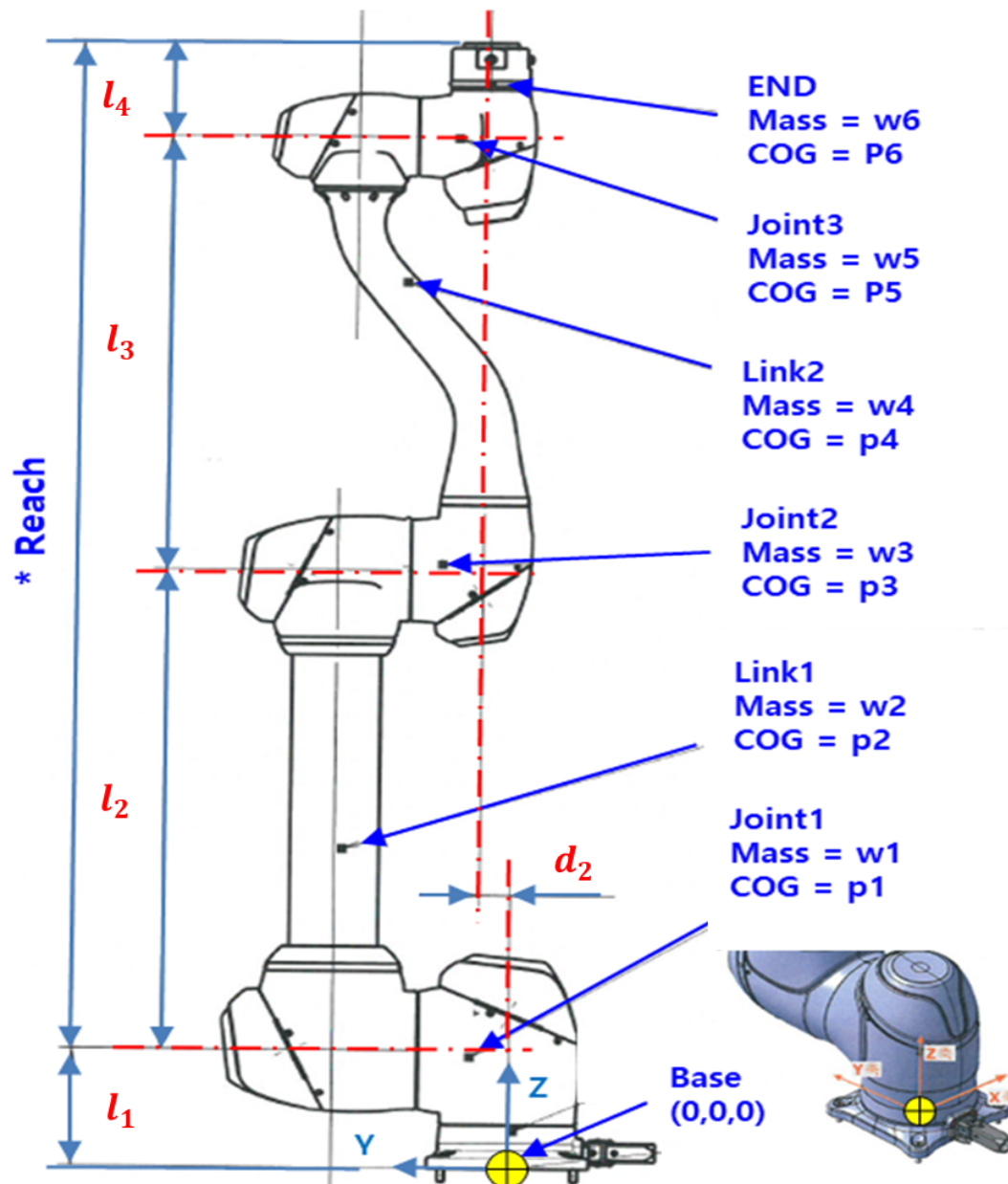
© 2022 Doosan Robotics Inc.

Doosan robot series(M/H/A)

Parameter and COG is defined as below figure

Caution

These values is only supported for your Calculation.



Defined The Denavit-Hartenberg Parameters for M/ H/ A-Series

Kinematic	theta [degree]	a [mm]	d [mm]	alpha [degree]
Joint1	0	0	l1	0
Link1	-90	0	d2	-90
Joint2	90	l2	0	0
Link2	0	0	l3	90
Joint3	0	0	0	-90
End	0	0	l4	90

DH Value of Each M & H-series : Unit [mm]

Value	M0609	M0617	M1013	M1509	H2017	H2515
l1	135	152.5	152.5	152.5	344.3	344.3
l2	411	845	620	411	845	759.5
l3	368	734	559	368	734	619.5
d2	6.25	34.5	34.5	34.5	9.9	9.9
l4	121	121	121	121	121	121

CoG Value Data of Each M & H-series : Point Value is defined by Base Coordination.

COG Value		M0609	M0617	M1013	M1509	H2017	H2515
Joint1	w1 [kg]	5.02	7.8	7.8	7.8	20.25	20.25
	p1 X	0.07	0.12	0.12	0.12	-0.05	-0.05

COG Value			M0609	M0617	M1013	M1509	H2017	H2515
	[m m]	Y	36.23	42.8	42.8	42.8	118.54	118.54
		Z	131.58	146.12	146.12	146.12	284.32	284.32
Link1	w2 [kg]		8.04	11.55	10.83	10.06	28.54	28.11
	p2	X	0.03	-0.05	-0.05	-0.06	-0.02	-0.02
	[m m]	Y	166.3	192.7	192.32	191.84	235.17	235.17
		Z	339.47	510.02	412.23	323.09	724.85	701.59
Joint2	w3 [kg]		3.6	3.68	3.68	3.68	6.42	6.42
	p3	X	-0.02	-0.015	-0.02	-0.02	-0.06	-0.07
	[m m]	Y	49.7	79.11	79.11	79.13	59.43	59.43
		Z	552.87	1004.2	779.2	570.21	1197.63	1111.21
Link2	w4 [kg]		3.57	4.28	3.82	3.59	7.59	7.3
	p4	X	0.04	0.057	0.07	0.04	0.08	0.08
	[m m]	Y	103.33	125.79	126.38	131.81	138.86	140.58
		Z	804.19	1478.31	1148.98	821.9	1702.74	1543.83
Joint3	w5 [kg]		2.83	2.8	2.82	2.83	4.1	4.1
	p5	X	-0.07	-0.27	-0.27	-0.07	-0.06	-0.06
	[m m]	Y	38.22	665.87	66.59	66.51	58.71	58.69
		Z	910.17	1727.85	1327.85	927.67	1923.89	1723.89

COG Value			M0609	M0617	M1013	M1509	H2017	H2515
END	w6 [kg]		1.16	1.16	1.16	1.16	0.9	0.9
	p6	X	-0.03	-0.29	-0.29	-0.03	-0.03	-0.03
	[m m]	Y	6.21	34.51	34.51	34.51	9.94	9.91
		Z	981.1	1798.6	1398.6	998.6	1994.86	1794.9

DH Value of Each A-series : Unit [mm]

Value	A0509	A0509S	A0912	A0912S
l1	155.5	155.5	195	195
l2	409	409	560	560
l3	367	367	516	516
d2	0	0	39	39
l4	124	127	124	127

CoG Value Data of Each A-series : Point Value is defined by Base Coordination

COG Value			A0509	A0509S	A0912	A0912S
Joint1	w1 [kg]		3.72	3.72	7.86	7.86
	p1	X	-0.069	-0.069	0.044	0.044
	[m m]	Y	24.423	24.423	29.982	29.982
		Z	148.125	148.125	183.88	183.88
Link1	w2 [kg]		6.84	6.84	10.99	10.99

COG Value			A0509	A0509S	A0912	A0912S
	p2	X	0	0	1.426	1.426
	[m m]	Y	132.71	132.71	170.908	170.908
		Z	359.986	359.986	392.484	392.484
Joint2	w3 [kg]		2.77	2.77	2.88	2.88
	p3	X	0.071	0.071	0.085	0.085
	[m m]	Y	33.009	33.009	71.058	71.058
		Z	569.623	569.623	757.513	757.513
Link2	w4 [kg]		2.68	2.68	3.27	3.27
	p4	X	-0.086	-0.086	-0.096	-0.096
	[m m]	Y	86.348	86.348	124.314	124.314
		Z	845.469	845.469	1131.255	1131.255
Joint3	w5 [kg]		2.05	2.05	2.14	2.14
	p5	X	0.091	0.091	0.109	0.109
	[m m]	Y	15.434	15.434	53.77	53.77
		Z	937.957	937.957	1274.286	1274.286
END	w6 [kg]		0.77	0.87	0.78	0.88
	p6	X	-0.022	-0.022	-0.019	-0.019
	[m m]	Y	-0.007	-0.007	38.994	38.994

COG Value			A0509	A0509S	A0912	A0912S
		Z	1003.754	1007.754	1342.73	1346.73

Bijlage B

Bijlage - Python-script voor het inlezen van de sensordata en opstellen van een map


```

import re
import matplotlib.pyplot as plt
import numpy as np

# Data containers
data = {
    "timestamps": {"A": [], "B": []},
    "wheel": {"A": {"L": [], "R": []}, "B": {"L": [], "R": []}},
    "velocity": {"A": {"L": [], "R": []}, "B": {"L": [], "R": []}},
    "current": {"A": {"L": [], "R": []}, "B": {"L": [], "R": []}},
    "encoder": {"A": {"pivot": [], "velocity": []}, "B": {"pivot": [], "velocity": []}},
    "torque difference": [],
    "torque": {
        "Voorwaarts": [],
        "Zijwaarts": [],
        "Links": [],
        "Rechts": [],
        "Voor": [],
        "Achter": []
    },
    "total_current": {
        "A": [],
        "B": [],
        "sum": []
    }
}

def parse_file(filename, time_unit='s'):
    with open(filename, 'r') as file:
        firstTimestampA = 0
        for line in file:

            timestamps_A = re.findall(r"Drive A - Timestamp:\s*([0-9]*\.[0-9]*e[+-]?[0-9]+)", line)
            timestamps_B = re.findall(r"Drive B - Timestamp:\s*([0-9]*\.[0-9]*e[+-]?[0-9]+)", line)

            if firstTimestampA == 0:
                firstTimestampA = float(timestamps_A[0]) / (1e9 if time_unit == 's' else 1e6)

            if timestamps_A:
                # Process both timestamps for A and B
                data["timestamps"]["A"].append((float(timestamps_A[0]) / (1e9 if time_unit == 's' else 1e6)) - firstTimestampA)

            elif timestamps_B:
                data["timestamps"]["B"].append((float(timestamps_B[0]) / (1e9 if time_unit == 's' else 1e6)) - firstTimestampA)

            # Now process the rest of the data
            match = re.findall(r"[-+]?[0-9]*\.[0-9]*e[+-]?[0-9]+", line)

            # --- Drive Data Parsing
            if "Drive A" in line or "Drive B" in line:
                drive = 'A' if "Drive A" in line else 'B'
                if "Wheel" in line and len(match) >= 2:
                    data["wheel"][drive]["L"].append(float(match[0]))
                    data["wheel"][drive]["R"].append(float(match[1]))
                elif "velocity" in line and "L" in line and len(match) >= 2:
                    data["velocity"][drive]["L"].append(float(match[0]))
                    data["velocity"][drive]["R"].append(float(match[1]))
                elif "current L" in line and len(match) >= 2:
                    data["current"][drive]["L"].append(float(match[0]))
                    data["current"][drive]["R"].append(float(match[1]))
                elif "encoder" in line and len(match) >= 2:
                    data["encoder"][drive]["pivot"].append(float(match[0]))
                    data["encoder"][drive]["velocity"].append(float(match[1]))

            # --- Total Current Fields
            elif "Drive A current" in line:
                data["total_current"]["A"].append(float(match[0]))
            elif "Drive B current" in line:
                data["total_current"]["B"].append(float(match[0]))
            elif "Total current sum" in line:
                data["total_current"]["sum"].append(float(match[0]))

            # --- Torque Fields
            elif "forward torque" in line:
                data["torque"]["Voorwaarts"].append(float(match[0]))
            elif "Sideways torque" in line:
                data["torque"]["Zijwaarts"].append(float(match[0]))
            elif "left torque" in line:
                data["torque"]["Links"].append(float(match[0]))
            elif "right torque" in line:
                data["torque"]["Rechts"].append(float(match[0]))
            elif "front torque" in line:
                data["torque"]["Voor"].append(float(match[0]))
            elif "back torque" in line:

```

```

        data["torque"]["Achter"].append(float(match[0]))

    # Legacy support
    elif "torque difference" in line:
        data["torque difference"].append(float(match[0]))

wheel_configs = [
    {
        "pivot_position": {"x": -0.7, "y": -0.225},
        "a": 4.794289,
        "relative_position_l": {"x": -0.01, "y": 0.0275},
        "relative_position_r": {"x": -0.01, "y": -0.0275},
        "linear_to_angular_velocity": 19.05,
        "reverse_velocity": False
    },
    {
        "pivot_position": {"x": -0.7, "y": 0.225},
        "a": 3.196517,
        "relative_position_l": {"x": -0.01, "y": 0.0275},
        "relative_position_r": {"x": -0.01, "y": -0.0275},
        "linear_to_angular_velocity": 19.05,
        "reverse_velocity": False
    }
]

def norm(x):
    TWO_PI = 2.0 * np.pi
    while x < -np.pi:
        x += TWO_PI
    while x > np.pi:
        x -= TWO_PI
    return x

def calculate_robot_velocity(data, wheel_configs, r_w=0.0525):
    vx_list, vy_list, va_list = [], [], []

    for i in range(len(data["timestamps"]["A"])):
        A = []
        b = []

        for j, wheel in enumerate(["A", "B"]):
            wl = data["velocity"][wheel]["L"][i]
            wr = -data["velocity"][wheel]["R"][i]
            pivot_angle_raw = data["encoder"][wheel]["pivot"][i]
            wp = data["encoder"][wheel]["velocity"][i]

            cfg = wheel_configs[j]

            # Apply pivot offset
            pivot_angle = norm(pivot_angle_raw - cfg["a"])
            unit_vec = np.array([np.cos(pivot_angle), np.sin(pivot_angle)])

            # Convert wheel angular velocities to linear (m/s)
            lin_to_ang = cfg["linear_to_angular_velocity"]
            vel_l = wl / lin_to_ang
            vel_r = wr / lin_to_ang

            # Average projection velocity
            v_proj = 0.5 * (vel_l + vel_r)

            # Reconstruct normal velocity (from wheel geometry)
            slope_l = cfg["relative_position_l"]["y"] / cfg["relative_position_l"]["x"]
            slope_r = cfg["relative_position_r"]["y"] / cfg["relative_position_r"]["x"]
            slope_avg = 0.5 * (slope_l + slope_r)
            v_normal = (vel_l - v_proj) / slope_l if abs(slope_l) > 1e-3 else 0.0

            # Reconstruct full pivot velocity vector
            v_pivot = unit_vec * v_proj + np.array([-unit_vec[1], unit_vec[0]]) * v_normal

            # Reverse if necessary
            if cfg["reverse_velocity"]:
                v_pivot *= -1

            # Module position
            xj = cfg["pivot_position"]["x"]
            yj = cfg["pivot_position"]["y"]

            # Motion constraint equations:
            A.append([1, 0, -yj]) # vx - va * yj = v_pivot_x
            b.append(v_pivot[0])
            A.append([0, 1, xj]) # vy + va * xj = v_pivot_y
            b.append(v_pivot[1])

        A = np.array(A)
        b = np.array(b)

```

```

        result, _, _, _ = np.linalg.lstsq(A, b, rcond=None)
        vx_list.append(result[0])
        vy_list.append(result[1])
        va_list.append(result[2])

    return vx_list, vy_list, va_list

def mapping(vx, vy, va):
    dt = 0.05
    x, y, theta = 0.0, 0.0, 0.0
    positions = [(x, y)]

    for i in range(len(vx)):
        dx = vx[i] * dt
        dy = vy[i] * dt

        # Mid-point angle approximation
        a_avg = -theta + 0.5 * va[i] * dt
        cos_a = np.cos(a_avg)
        sin_a = np.sin(a_avg)

        x += dx * cos_a - dy * sin_a
        y += dx * sin_a + dy * cos_a
        theta += va[i] * dt

        positions.append((x, y))

    positions = np.array(positions)

    plt.scatter(positions[:, 0], positions[:, 1], s=5, c='blue')
    plt.xlabel("X (m)")
    plt.ylabel("Y (m)")
    plt.title("Platform Path (Dead Reckoning from Odometry)")
    plt.axis("equal")
    plt.grid(True)
    plt.show()

#=====
parse_file('interaction_5/interaction_data_turn.txt', time_unit='s')
#=====

vx, vy, va = calculate_robot_velocity(data, wheel_configs)

mapping(vx, vy, va)

```