

**Ilkka Heikkinen**

# **VOIMAOHJAUKSEN HYÖDYNTÄMINEN UR3-ROBOTILLA**

**Opinnäytetyö  
CENTRIA-AMMATTIKORKEAKOULU  
Tuotantotalouden koulutus  
Maaliskuu 2019**

<b>Centria-ammattikorkeakoulu</b>	<b>Aika</b> Maaliskuu 2019	<b>Tekijä/tekijät</b> Ilkka Heikkinen
<b>Koulutusohjelma</b> Tuotantotalous, Insinööri		
<b>Opinnäytetyön nimi</b> Voimaohjauksen hyödyntäminen UR3 Robotilla		
<b>Työn ohjaaja</b> Jari Kaarela, Sakari Pieskä		<b>Sivumäärä</b> 45 + 4
<b>Työelämäohjaaja</b> Jari Mäkelä		
<p>Tässä työssä tarkoituksena oli tutustua yhteistyörobotiikkaan ja voima-anturien toimintaan ja niiden tuomiin lisämahdollisuuksiin tuotannossa. Lisäksi haluttiin tutustua ArtiMinds-robotinohjelmointiohjelmaan koska koulu oli ostanut vasta sen lisenssin ja sitä ei oltu juurikaan käytetty. Työn tuloksia on tarkoitus käyttää Centria-ammattikorkeakoulun oman tutkimus-, kehitys- ja innovaatiotoimintaan.</p> <p>Työn suunnittelun ja aikatauluttamisen jälkeen alkoi ensin teoriaan ja laitteistoon tutustuminen. Samaan aikaan aloitin teoriaosioden kirjoittamisen, jotta tämän työn lukija ymmärtäisi paremmin työn eri vaiheet ja työn tulokset.</p> <p>Työssä piti ensin tehdä SolidWorks-ohjelmalla 3D-malli työpisteestä, jossa robotin toimintaa tutkitaan. Mallin valmistuttua se liitettiin ArtiMinds-ohjelmalle, missä voitiin tehdä sitten kyseisten ohjelmien simulaatiot. Mallia ei varsinaisesti tarvinnut ohjelmakoodien tekoon ArtiMindsilla, mutta käyttäjä tarvitsee sen varsinkin törmäysten estämisen varmistamiseksi.</p> <p>Kokeilin työn aikana ohjelmakoodien tekoa sekä ilman mallia että mallin kanssa. Kokeilin myös eri toimintojen vaikutusta robotin varsinaiseen toimintaan. Työssä tehtiin kaksi erilaista demonstraatiota, joiden tarkoituksena oli kuvastaa näiden ohjelmien tapaa koota ohjelmakoodeja samalla hyödyntäen OptoForcen 6-akselin voima-anturin antamia mittatietoja. Ensimmäinen on palletointiin sopeutuva ohjelma ja toinen ohjelma demonstroi kokoonpanoon sopeutuvaa ohjelmaa.</p> <p>Lopuksi kävin läpi työn tulokset ja tein oman pohdintani työssä tutkittavista asioista. Kävin tässä läpi omien kokemusten perusteella asiasta tehtyjä havaintoja sekä arvioita näiden uusista tulevaisuuden mahdollisuuksista ja niiden vaikutuksista.</p>		
<b>Asiasanat</b> ArtiMinds, Robotiikka, UR3, Voima-anturi, Yhteistyörobotiikka		

<b>Centria University of Applied Sciences</b>	<b>Date</b> March 2019	<b>Author</b> Ilkka Heikkinen
<b>Degree programme</b> Industrial Management, Engineer		
<b>Name of thesis</b> Utilizing Force Control with the UR3 Robot		
<b>Instructor</b> Jari Kaarela, Sakari Pieskä		<b>Pages</b> 45 + 4
<b>Supervisor</b> Jari Mäkelä		
<p>The goal of this thesis was to find out what kinds of applications force control can bring to robotics. The main focus was on collaborative robots. Another goal was to get to know the ArtiMinds RPS robot programming software, to which my school had just bought all the licenses. The results of this thesis will be used at the school of Centria University of Applied Sciences R&amp;D department.</p> <p>After scheduling and making the project plan for this work, I began to do my research for the theory parts. At the same time, I started to write the theory parts. I wrote them so anyone who reads this could understand the results of this work more clearly.</p> <p>First, I had to make a 3D-model of the real workstation. I made it with the SolidWorks CAD software. After the model was made, it could be applied to the ArtiMinds RPS. One does not actually need it to make the programs, but I needed it to make sure the program was completely collision free.</p> <p>I tested making the programs with and without the 3D-model. I also tested the different Template models and their actions, and how they affected the movement of the actual robot. In this thesis there are two different program codes. I made. During the programming I used the OptoForce F/T sensor to help parametrize the programs. The first one is a palletizing program which is applicable for automation. The second one is a demonstration of a force controlled assembly.</p> <p>Finally the results are presented as well as my speculation of the results. I discuss different observations I made while making these programs and testing the different Templates. I also discuss some possible effects of collaborative robots on the future.</p>		

<b>Key words</b> ArtiMinds, Collaborative Robots, Force Torque -Sensor, Robotics, UR3
--

## KÄSITTEIDEN MÄÄRITTELY

ArtiMinds	Robotin ohjelmointiin tarkoitettu saksalaisvalmisteen ohjelmointiohjelma, joka käyttää ohjelmointiin pääosin imitointia
CAD	Tulee sanoista Computer-aided Design, joka siis tarkoittaa tietokoneavusteista suunnittelua. CAD-työkaluja käytetään varsinkin arkkitehtien ja insinöörien suunnittelutehtävissä, jotka sisältävät numeerista laskentaa, 2D-piirtämistä, 3D-mallinnusta ja tietokonesimulointia.
IAD	Intelligent Assist Device – robotin tyyppinen apulaite, joka auttaa työntekijöitä nostamaan taakkoja
IFR	International Federation of Robotics = Kansainvälinen Robottiyhdistys
IoT	Internet of Things = Esineiden internet
ISO	International Organization for Standardization
LiDAR	Light Detection and Ranging on optinen tutka, joka toimii näkyvän valon, lähi-infran tai ultravioletin alueella
Ohjelmakoodi	Puhutaan myös pelkästä koodista tai ohjelmasta. Tarkoitetaan sitä koodia, joka koostuu kokonaisuudesta eri toimintoja, jotka muodostavat halutun tehtävän. Puhun tässä työssä ohjelmakoodista sen takia, että käsitteet ohjelmisto ja koodi eivät menisi sekaisin.
Pk-yritys	Pieni ja keskisuuri yritys. Vähemmän kuin 250 työntekijää ja vuosittainen liikevaihto korkeintaan 50 miljoonaa euroa tai taseen loppusuma on enintään 43 miljoonaa euroa.
RPS	Robot Programming Suite – ArtiMinds-ohjelmiston nimi

TKI	Tutkimus, kehitys ja innovaatio
TÜV	Technischer Überwachungsverein saksalainen yritys, joka on erikoistunut tuotteiden tarkastukseen ja sertifiointiin.
UR3	Tanskalaisen Universal Robots -yrityksen valmistama yhteistyörobotti
Voima-anturi	Voiman monipuoliseen mittaamiseen tarkoitettu anturi, joka antaa robotille “tuntoaistin”

# SISÄLLYSLUETTELO

## TIIVISTELMÄ

## ABSTRACT

## KÄSITTEIDEN MÄÄRITTELY

<b>1 JOHDANTO</b> .....	<b>1</b>
<b>2 ROBOTIIKKA</b> .....	<b>3</b>
2.1 Teollisuusrobotit .....	5
2.2 Yhteistyörobotit .....	7
<b>3 VOIMAOHJAUS</b> .....	<b>11</b>
<b>4 LAITTEET JA OHJELMISTOT</b> .....	<b>14</b>
4.1 UR3-robotti .....	14
4.2 ArtiMinds -ohjelmointiohjelma .....	16
4.2.1 Template -mallit .....	19
4.2.2 Wizards -apuohjelmat .....	20
4.2.3 Visualisointi ja simulointi .....	21
4.2.4 Esimerkki ohjelmakoodin luomiseen .....	21
4.3 OptoForce voima-anturi .....	22
4.4 SolidWorks 3D CAD -suunnitteluohjelma .....	24
<b>5 TYÖVAIHEET / TYÖN ETENEMINEN</b> .....	<b>26</b>
5.1 Palletointi .....	26
5.2 Kokoonpano ja asennustyö BRIO-palikoilla .....	34
<b>6 TULOKSET</b> .....	<b>37</b>
<b>7 POHDINTA</b> .....	<b>38</b>
<b>LÄHTEET</b> .....	<b>43</b>
<b>LIITTEET</b>	
<b>TAULUKOT</b>	
TAULUKKO 1. Ennuste yhteistyörobottien myynnin kehityksestä	8
TAULUKKO 2. OptoForce HEX-70-XE-200N Tekniset tiedot	24
TAULUKKO 3. Ennuste yhteistyörobottien eri applikaatioalojen myyntituloista vuoteen 2022 mennessä	41
<b>KUVIOT</b>	
KUVIO 1. Voimaohjauksen säätökaavio	12
<b>KUVAT</b>	
KUVA 1. Teollisuuden aikakaudet/vallankumoukset	4
KUVA 2. George Devol'in patenttihakemus	6
KUVA 3. UR3-robotti	14

KUVA 4. Robotin tabletti-ohjain	16
KUVA 5. ArtiMinds RPS -oletusnäkyvä	19
KUVA 6. OptoForcen 6-akselinen voima-anturi	23
KUVA 7. Robotin kokoaminen Configure kohdassa.	27
KUVA 8. ArtiMinds RPS:n simulaation 3D näkyvä palletoinnista ilman 3D-mallia.	29
KUVA 9. Kuva varsinaisesta työpisteestä.	30
KUVA 10. SolidWorksin 3D CAD-malli	31
KUVA 11. Collisions havainnointi 3D näkyvässä	32
KUVA 12. ArtiMinds RPS näkyvä 3D-mallin kera ja ohjelmakoodin pisteet nähtävissä	33
KUVA 13. BRIO-kotelo asennettuna kulmaan ja eri palikat kotelon edessä omilla noutopisteillä	35

## 1 JOHDANTO

Tämä työ on Centria-ammattikorkeakoulun tuotantotalouden AMK, insinööritutkinnon opinnäytetyö. Tässä työssä tarkoituksena oli tutustua yhteistyörobotiikkaan ja voima-anturien toimintaan sekä niiden tuomiin lisämahdollisuuksiin tuotannossa. Tavoitteena oli tutustua tarkemmin ArtiMinds-ohjelmaan ja sillä ohjelmoimiseen sekä tutkia voima-anturin tuomia lisäapplikaatioita. ArtiMinds on uusi robotin ohjelmointiin tarkoitettu ohjelma ja sen lisenssi oli vasta hankittu koululle, joten sen toiminta ja mahdollisuudet eivät olleet Centrian henkilökunnalla vielä tiedossa. Työni tuloksia on tarkoitus hyödyntää Centrian omaan tutkimus-, kehitys- ja innovaatiotoiminnassa. Tutkittavana kohteena olivat voima-anturin tuomat lisämahdollisuudet ja eritoten voimaohjaus. Tärkeimpänä lähdemateriaalina toimi ArtiMinds RPS Expert Training -kirja, (Jäkel, Schmidt-Rohr, Stein 2016) Universal Robots:in internetsivut ja heidän julkaisut (Universal Robots 2019) sekä Erlbacherin voimaohjauksen perusteet (Erlbacher 2000).

Tarkoituksena työssä oli tehdä UR3- pöytätason yhteistyörobotille ArtiMinds -ohjelmointiohjelmalla palletointiohjelma ja sekä tutkia että demonstroida OptoForcen voima-anturin toimintaa. Voima-anturilla pitäisi pystyä varmistamaan kappaleen kunnollinen kiinnittyminen ja estää liiallinen voimankäyttö vahinkojen estämiseksi sekä mahdollistaa monet muutkin sovellukset, kuten esimerkiksi voimaohjaus. Nämä sovellukset vaativat kolmen eri suunnan voiman sekä kolmen vääntömomentin mittauksen. Tämä kaikki tulisi saada toimimaan etäohjelmoinnilla. Työ tehtiin Centria-ammattikorkeakoulun Ylivieskan kampuksella tuotantoteknologian laboratoriossa, jossa UR3-robotti oli valmiiksi asennettuna työpisteen pöytään. Työpisteen viereen oli kaapeleilla liitettyä kannettava tietokone, johon oli valmiiksi asennettu ArtiMinds RPS.

Työssä piti ensin tehdä työpisteestä SolidWorks-ohjelmalla 3D-malli, jolla tutkitaan robotin toimintaa ja mahdollistetaan parempien ohjelmien teko. 3D-mallin valmistuttua se voitiin liittää ArtiMinds RPS:lle, jolla suoritettiin kyseisen ohjelman simulaatiot. Mallia ei varsinaisesti tarvita ohjelmakoodin tekoon ArtiMindsilla, mutta käyttäjä tarvitsee sitä varsinkin törmäysten estämisen ehkäisemiseksi.

Mallinnuksen jälkeen alkoi varsinaisten ohjelmakoodien teko. Kokeilin myös ohjelmakoodin tekoa ilman mallia, ennen mallin liittämistä ArtiMindsille. Opinnäytetyössä tehdyt robotin ohjelmat soveltuvat automaatio- ja kokoonpanotehtäviin. Automaation osuutta demonstroi palletointi ja



kokoonpanoa voimaohjauksella tehty asennus. Tavoite oli luoda toimivat ohjelmat, jotka demonstroivat voima-anturin luomia lisämahdollisuuksia.

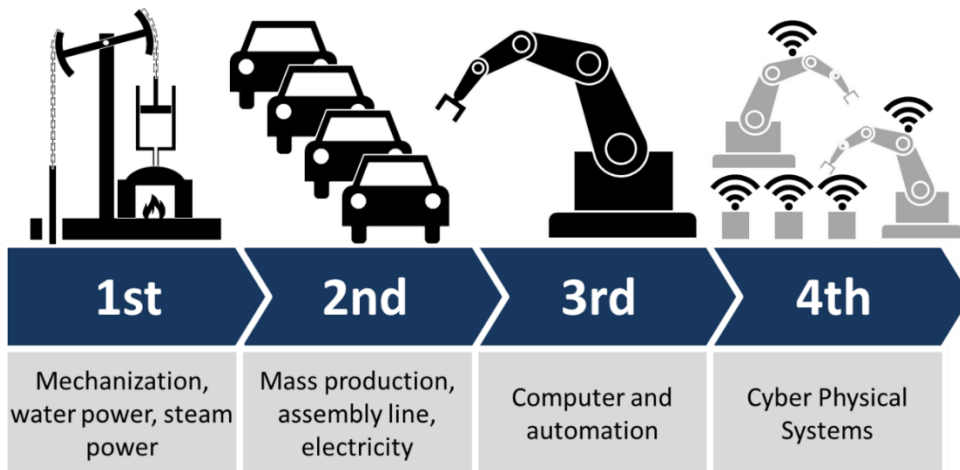
## 2 ROBOTIIKKA

Robottiikalla tarkoitetaan tieteen ja tekniikan alaa, jossa yhdistyvät useiden eri alojen tietämys ja osaaminen. Robottien suunnitteleminen, valmistaminen ja soveltaminen ovat robotiikan alan päätarkoitus. (ISO 8373 2012 2.16) Alalla tarvitaan tietämystä mekaniikasta, ohjelmoinnista sekä säätö- ja konetekniikasta. Viime aikoina robotiikan kehitys on ollut yhä enemmän esillä ja roboteista on tullut osa arkipäivää monissa tuotannoissa. Ilman robotiikan mahdollistamaa automatisointia monen yrityksen kilpailukyky ei voisi olla sitä luokkaa kuin mitä se nyt on. Kaikesta tästä huolimatta tärkein tekijä robotiikan hyödyntämisessä on kuitenkin ihminen. Ihminen suunnittelee ja kokoaa järjestelmät sekä ohjelmoi laitteet ja pitää ne kunnossa. (Salminen 2016)

Ennen robotiikkaa kappaleenkäsittely automatisoiduissa sovelluksissa oli hyvin kankeasti suoritettu hydraulisesti tai pneumaattisesti. Koneet olivat sovelluskohtaisia ja niissä ei ollut minkäänlaista joustoa muutosten suhteen. Nämä olivat tarkoitettuja hyvin suuriin tuotantoeriin, ja pienetkin muutokset asetuksiin saattoi vaatia laitteiston täydellisen uusimisen. Tällainen tuotanto valmisti yleensä pitkälle vakioituja tuotteita suuria määriä varastoon. Ajan myötä joustavuus, asiakaslähteisyys ja piensarjatuotanto alkoivat olla yhä useammin vaatimuksena tuotannolle ja se kasvatti tarvetta ohjelmallisesti muunneltaville kappaleenkäsittely- ja toimilaitteille. 1980- ja 1990-luvuilla kehitetyt robotit vastasivat joustavuudeltaan aikaisempia mekaanisia toimilaitteautomatisointeja. Robotit olivat hankalia ja työläitä ohjelmoida, joten ne eivät soveltuneet hyödynnettäväksi piensarjatuotantoon. (Wallén 2018, 11-12)

Nykyiset robotit ovat luotettavuudeltaan ja käyttövarmuudeltaan hyviä. Robottien ohjelmointi on kehittynyt ripeästi uusien korkeatasoisten ohjelmointikielten ansiosta. Ne nopeuttavat ja helpottavat ohjelmakoodien tekoa, mikä on luonut tuotantoon uudenlaista joustoa. Jousto on nykyään roboteille olennaista, koska ohjelmoinnin lisäksi niiden on osattava muodostaa liikeratojaan sekä suunnittelun että ympäristömallin tiedoista. Robotteihin on mahdollista saada näitä ja muita lisäominaisuuksia liittämällä niihin erilaisia antureita, jotka mahdollistavat robotille kyvyn reagoida ympäristöönsä automaattisesti ja älykkäästi. Näiden antureiden ansiosta robotiikalle saadaan lisää soveltuvuuskohteita ja kyky automatisoida yhä vaativampia tehtäviä. Nykyään robotteja voidaan käyttää osana erilaisia laajoja tuotantojärjestelmiä. (Salminen 2016)

Tuotantojärjestelmät ovat kehittyneet näiden anturien ansiosta uusiin ulottuvuuksiin, sillä modernit anturit mahdollistavat IoT:n eli esineiden Internetin. Esineiden Internetin on sanottu tuovan teollisuuden neljännen vallankumouksen eli niin sanotun Teollisuus 4.0 (Industry 4.0 Wiki 2019). Teollisuuden eri vallankumoukset ovat nähtävillä kuvassa 1. Nykyisten pilvipalvelujen, datakeskusten ja prosessointitehon ansiosta antureilla voidaan hyödyntää internetin mahdollisuuksia laitteissa ja koneissa. Kansainvälisen ICT-alan konsultointiyhtiön Gartnerin määritelmän mukaan teollisessa Internetissä on kyse fyysisistä laitteista, jotka pystyvät aistimaan ympäristöään ja viestimään tai toimimaan aistimansa perusteella älykkäästi keskenään tai yksin. Tähän tarvitaan antureita, ohjelmistoja sekä tietoliikenneyhteys, jolloin anturit, koneet, prosessit ja palvelut tuottavat jatkuvasti tietoa, jota jalostamalla voidaan muun muassa ennakoita ja automatisoida eri tuotannonprosesseja ja muita työvaiheita. (Esineiden Internet Wikipedia 2019)



KUVA 1. Teollisuuden aikakaudet/vallankumoukset (Industry 4.0 Wikipedia 2019)

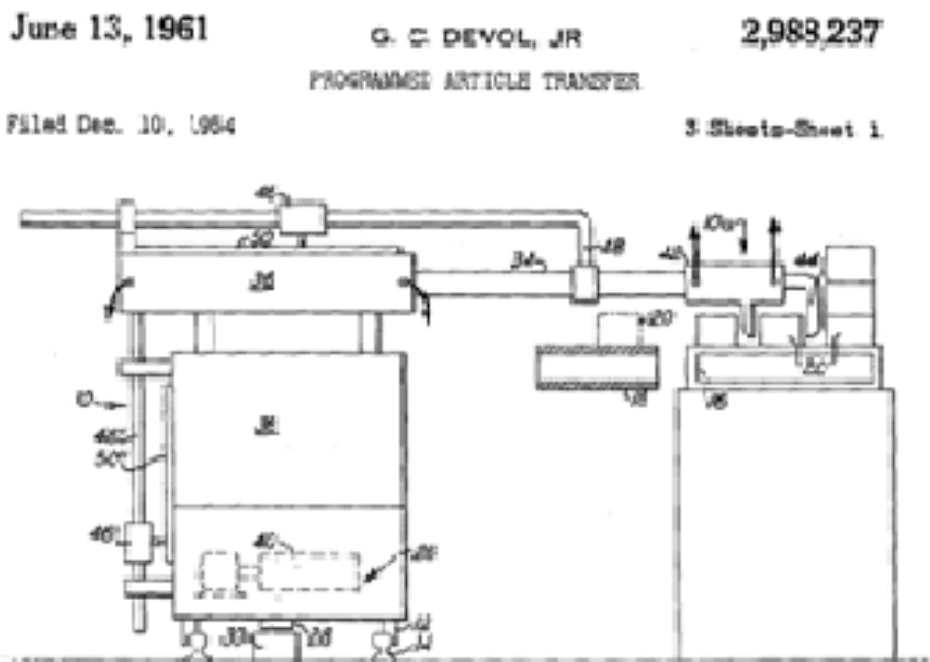
Teollisen Internetin järjestelmissä anturit siis keräävät tietoa esimerkiksi useista yksittäisistä sylintereistä tai roboteista, ja kaikki näiden toimilaitteiden tekemät liikkeet tallentuvat datana suoraan pilvitalennustilaan, minkä jälkeen ohjelma laskelmoi ja vertaa tietoja jo olemassa oleviin tietoihin. Tällä tavalla on mahdollista selvittää esimerkiksi pienetkin mahdolliset toimintaviat, ennen kuin virheellisesti toimiva laite hajottaa itseään tai sen vaikutusalueena sisällä olevia osia. Näiden tietojen perusteella voidaan harjoittaa muun muassa tuottavaa käynnissäpitoa ennakoimalla kaikki mahdolliset huollot ja tapaturmat. Ohjelma oppii myös yleisimmät vikojen syyt ja pystyy saamansa datan perusteella ehdottamaan mahdollisia toimenpiteitä. Ennakoiva kunnossapito ehkäisee kalliita seisokkeja ja muitakin hävikkejä (Industry 4.0 Wikipedia 2019)

Teollisen Internetin tuottama lisäarvo perustuu siihen dataan, jota se tuottaa keräämästään tiedosta. IoT:n avulla voidaan jalostaa prosessit tietynlaisiksi, jotta niiden tuottama tieto on helposti automaattisen tietoverkoston käytettävissä. Tietoverkosto kokoaa valtavan datamassan helposti tarkasteltavaan muotoon erilaisina tunnuslukuina. Prosessit keräävät siis raakadataa, jota analysoimalla saadaan reaaliaikaista tietoa niin laitteista kuin koneista. Tarkoituksena on luoda datan avulla ratkaisuja, jotka tuovat tulosta eivätkä pelkästään lisää dataa. Tämä mahdollistaa poikkeamien havaitsemisen ja ennakoinnin reaaliajassa, mikä estää taloudellisia vahinkoja ja pienentää riskiä monin eri tavoin. Monet teollisuuden alat hyödyntävät jo ennakoivaa analytiikkaa, josta esimerkkinä on palveluiden hinnan optimaalinen määrittely maksimaalisen voiton takaamiseksi. Parhaan hyödyn saa automatisoimalla tietojärjestelmän analytiikan, koska se tuottaa jatkuvasti reaaliaikaista ja ennakoivaa dataa. Sen avulla yritys voi tehostaa helpommin tuotantoaan ja ohjaamaan omia resurssejaan tehokkaammin oikeisiin paikkoihin. Tämä kaikki laskee oman toiminnan kustannuksia, vähentää hukkaa sekä parantaa yrityksen materiaali-, informaatio- ja rahavirtausten toimintaa. IoT:n on muun muassa ennustettu tavoittavan 2020 mennessä globaalisti 7,1 biljoonan dollarin markkinat, joka koostuu 30 miljardista verkkoon liitetystä laitteesta. Suomeenkin on kaavailtu IoT:n kautta 48 000 uutta työpaikkaa. (Esineiden Internet Wikipedia 2019)

## 2.1 Teollisuusrobotit

Robotin määritelmäksi ei ole oikeastaan olemassa yhtenäistä yksiselitteistä kuvausta. Kansainvälisen robottiyhdistyksen IFR:n mukaan robotti on vähintään kolminivelinen mekaaninen laite, joka on uudelleen ohjelmoitavissa. Sen täytyy olla suunniteltu liikuttamaan kappaleita, osia, työkaluja tai erikoislaitteita ohjelmoitavin liikkein monenlaisten tehtävien suorittamiseksi teollisuuden sovelluksissa. (IFR 2019) Uudelleen ohjelmointi on erittäin olennainen osa nykyisiä robotteja. ISO-standardisointijärjestön mukaan robotin määritelmä on “Teollisuuden automaatio-sovelluksissa käytettäväksi tarkoitettu automaattisesti ohjattu, uudelleen ohjelmoitavissa oleva monikäyttöinen käsittelylaite, jonka akseleista vähintään kolme on ohjelmoitavissa ja joka voi olla kiinteästi asennettu tai liikkuva” (ISO 10218-1 2011). Vaikka nämä määritelmät muistuttavat suuresti toisiaan, siitä huolimatta Japanissa manipulaattorit ja yksinkertaiset kappaleen siirtokoneet lasketaan myös roboteiksi. Yksinkertaistettuna teollisuusrobotti on mekaaninen kone, joka liikuttaa kiinnitys-laippaa halutulla tavalla, jonka liikkeet voidaan tuottaa sähköisten, pneumaattisten tai hydraulisten toimilaitteiden avulla. (Salminen 2016)

Ensimmäiselle teollisuusrobotille jätettiin patenttihakemus amerikkalaisen George Devol'in toimesta vuonna 1954. Patenttihakemuksen nimi oli Programmed Article Transfer ja se myönnettiin vasta 1961 (KUVA 2). Sen pohjalta rakennettiin prototyyppi, jonka nimeksi tuli Unimate#001. Robotti otettiin käyttöön General Motors'in tuotannossa vuonna 1961. Robotin tehtävänä oli nostaa painevalukoneesta kuumat metallilevyt pois ja pinota ne yhteen kasaan. (Stone 2005) Myöhemmin vuonna 1969 General Motors laajensi robottien käyttöä ja alkoi hyödyntää niitä pistehitsauksessa. (Wallén 2008, 10) 1960- ja 1970-luvulla robotit eivät vielä yleistyneet, koska niiden avulla ei vielä saavutettu suuria taloudellisia etuja. Näihin aikoihin kuitenkin nykyiset suuret robottivalmistajat alkoivat suunnittelemaan ja valmistamaan robottejaan. Uusia ominaisuuksia kehitettiin ja tehtiin merkittäviä parannuksia mekanismeihin ja toimintoihin. Kuuden eri akselin ja sitä suurempien vapausasteiden käyttö mahdollistui ja ohjausjärjestelmät alkoivat kehittyä tietokonepohjaiseksi. 1970-luvulla valmistuivat muun muassa ensimmäinen ihmismäinen WABOTI-robotti sekä SCARA-robotti, jonka rakenne on vieläkin laajasti teollisuuden käytössä (Salminen 2016)



KUVA 2. George Devol'in patenttihakemus (Cybeneticzoo 2013)

1980- ja 1990-luvun aikaan teollisuusrobottien kustannukset laskivat ja mahdollisuudet laajenivat. Yksitoikkoiset toistot, tarkat toleranssit ja haastavat olot alkoivat olla yleisiä robotin sovelluksia. Yksi merkittävä käyttökohde oli myös liukuhihnatuotannon nopeuttaminen. Robotit saatiin myös erikoistumaan paremmin erilaisiin yksittäisiin tuotannon työtehtäviin. (Wallén 2018, 11-12) Antureita ja konenäköä hyödyntäviä robottijärjestelmiä alkoi myös ilmestyä markkinoille. Vuonna

1985 General Motors muun muassa kehitti ohjelmoitavan robotin, jossa oli infrapuna- ja törmäysanturit, kamera- ja äänilyhteys, sekä äänisyntensaattori. Robotti oli lisäysten takia kykenevä toimimaan itsenäisesti ja oppimaan ympäristönsä (Salminen 2016)

Teollisuudessa roboteilla pyritään vähentämään tuotannon kustannuksia, lisäämään tuottoa ja parantamaan laatua tarkkuuden avulla. Teollisuuden sovellusten lisäksi robotteja kehitetään lääketieteen-, pelastus- ja etsintätehtävien, sekä viihde- ja palvelutekniikan käyttöön. (Wallén 2008, 14-15.) Hoito- ja palvelurobotteja on otettu jo joissain vanhainkodeissa ja ravintoloissa käyttöön. Muun muassa Ranskalaisen Robosoftin Kompai-robotti on erikoistunut näiden palvelutehtävien automatisointiin. (Spectrum 2010) Kompai-robotin myynti ja kehitys on nykyisin Kompai Robotics yrityksellä (Kompai Robotics 2019).

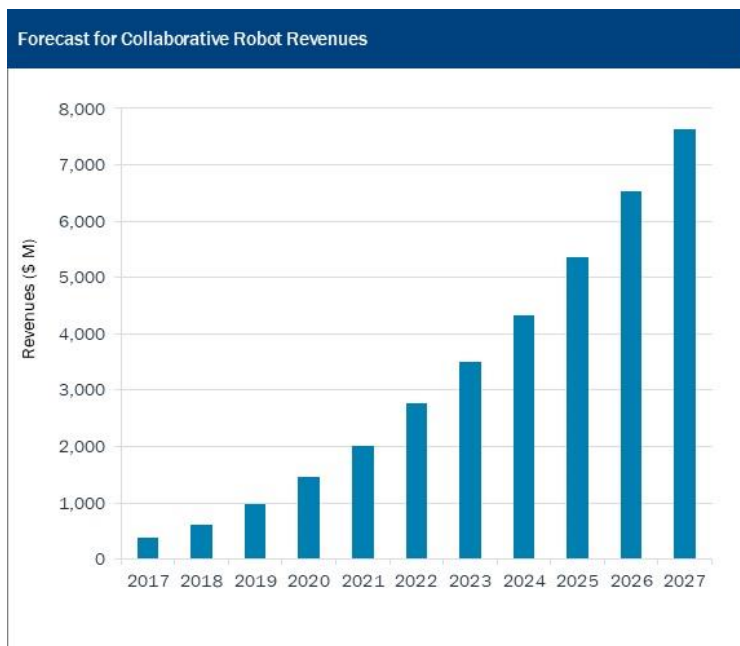
## **2.2 Yhteistyörobotit**

Ennen yhteistyöroboteja tuotannossa olleet robotit olivat yleensä työskennelleet turva-aidan sisällä erillään ihmisistä. Nykyään turvallisuusteknologian kehittyminen on mahdollistanut ihmisten ja robottien työskentelytilojen yhdistämisen. Tämä on nostanut tuotannon kapasiteettia, koska nyt robotit voivat toimia samassa tilassa ilman erillisiä turvatoimenpiteitä turvallisuutta kuitenkin laskematta. Tällaisista yhteistyöroboteista käytetään termiä cobot, joka tulee sanoista Collaborative Robot. (Cobot Wikipedia 2019)

Amerikkalainen General Motors oli myös edelläkävijä yhteistyörobotiikassa. Se alkoi etsiä ratkaisuja heidän tuotannon tarpeelleen saada tarpeeksi turvallisia robotteja, mitkä voisivat työskennellä ihmisten kanssa ilman erillistä turva-aitaa. General Motors teki yhteistyötä Professoreiden J. Edgar Colgaten ja Michael Peshkinin kanssa, mikä johti 1995 IAD:n (Intelligent Assist Device) syntyyn. IAD oli tietokoneen ohjaama turvallisuusväline, joka voiman tunnistamalla ohjasi työkuormaa työntekijän liikkeiden mukaisesti, täten pienentäen ihmisen taakkaa. (Pittman 2016) IAD oli ensimmäisiä kehitettyjä yhteistyöroboteja, mutta niistä ei vielä käytetty termiä cobot. Vuonna 1997 J. Edgar Colgate ja Michael Peshkin olivat ensimmäiset, jotka jättivät patenttihakemuksen nimenomaan yhteistyörobotista. Patentti hyväksyttiin vuonna 1999. (United States Patent 1999) Nämä olivat vielä kuitenkin alkukantaisia versioita, koska lähinnä piti olla jatkuvassa ihmisen ohjauksessa, eikä niiden mekaniikka tai autonomia ollut nykyisten robottien tasolla.

Ensimmäinen nykypäiväisiä yhteistyörobotteja muistuttava cobot LBR 3 tuli markkinoille vuonna 2004 saksalaisen KUKA:n toimesta. Vuonna 2008 Tanskalainen Universal Robots toi markkinoille ensimmäisen UR5-robottinsa. Tämä vei yhteistyörobottien kehitystä paljon eteenpäin ja tällöin syntyi nykyinen käsitys yhteistyörobotiikasta. Vaikka nämä robotit olivat pieniä, niiden työkuormat pystyivät olemaan yllättävänkin isoja. Roboteissa alkoi olla useita niveliä ja niiden autonomia parani. Monet robottivalmistajat seurasivat Universal Robots:in esimerkkiä ja nykyään markkinoilla onkin monenlaisia ja kokoisia yhteistyörobotteja eri valmistajilta. (Cobot Wikipedia 2019) Vuonna 2018 yhteistyörobottien markkinat räjähtivät ja kasvoivat edelliseen vuoteen verrattuna 60% nostaen globaalit myyntitulot noin 500 miljoonaan dollariin. Vuoteen 2027 mennessä näiden tulojen odotetaan nousevan 7,6 miljardiin dollariin. (Robotics Tomorrow 2018) Ennuste myyntituloista on nähtävissä taulukosta 1. Tällä hetkellä suurimpia toimijoita robottimarkkinoilla on FANUC KUKA ja ABB. (TechNavio 2018)

TAULUKKO 1. Ennuste yhteistyörobottien myynnin kehityksestä (Robotics Tomorrow 2018)



Teollisuusrobotit ovat enemmän suureen massatuotantoon, jotka tekevät paljon samanlaisia tuotteita, joiden tuotantosarjat ja tehtävät eivät juurikaan muutu. Yhteistyörobotit ovat mahdollistaneet sen, että myös pienemmät toimijat ovat voineet ottaa robotiikkaa mukaan omaan tuotantoon. Niiden uudelleen ohjelmointi on hyvin yksinkertaista ja nopeaa, joten se soveltuu myös paljon vaihtelua sisältäviin tuotantoihin. Näiden robottien avulla on helppo vaihtaa esimerkiksi vain tietty tuotannon osa ilman,

että koko tuotantolinjaa joudutaan purkamaan. Kehittyneet anturit ja turvajärjestelmät takaavat helpon siirrettävyyden ja mukautuvuuden, sekä poistavat tarpeet erillisille turvajärjestelmille. Joustavuus on yhteistyörobottien merkittävimpiä etuja.

On kuitenkin hyvä huomioida, että tarvittavien turvavaatimusten noudattaminen riippuu siitä, mitä tehtävää yhteistyöroboteilla halutaan suorittaa. Tästä syystä tulisi muistaa seuraavat kolme asiaa, jotka TM Robotics listasi yleisimmiksi virheellisiksi käsityksiksi yhteistyörobotiikasta (TM Robotics 2017, 2-3):

1. Väite, että yhteisen työtilan ihmisen kanssa voivat jakaa vain niin sanotut yhteistyörobotit, on virheellinen. Robotti itsessään ei takaa turvallista sovellusta. Lähes jokainen robotti taas on riittävin turvallisuusmekanismein kyvykäs yhteistyöhön, jos se täyttää standardin ISO 10218-1:2011 vaatimukset. Suuntaviivat turvallisen sovelluksen suunnitteluun löytyvät ISO:n teknisestä spesifikaatiosta ISO/TS15066:2016.

2. Se, että yhteistyörobotti ei koskaan tarvitse turva-aitaa ympärilleen, on myös virheellinen käsitys. Jokainen suunniteltu automatisoitu sovellus, jossa ihmisiä toimii samassa tilassa, tulee ennen toiminnan aloittamista tehdä riskianalyysi, jossa määritellään erilaisten turvaratkaisujen tarve. On täysin sovelluskohtaista, tarvitaanko turva-aitauksia vai ei. Kuten edellä mainittiin, robotti on vain yksi osa sovellusta, jonka turvallisuutta tarkastellaan aina kokonaisuutena.

3. Yhteistyörobotin kyvystä työskennellä aina ihmistä selvästi nopeammin ja tehokkaammin on myös harhakuvitelmia. Ihmisen ja robotin jakama työskentelyalue johtaa yhteentörmäyksen riskin kasvamiseen. Tällaisen kontaktin ihmiselle vahinkoa aiheuttavaa vaikutusta pienennetään robotin liikenopeutta vähentämällä tuottavuuden kustannuksella. Yhteistyörobotin ohjelmointi taas esimerkiksi ”taluttamalla” voi maallikon kannalta olla yksinkertaisempaa, mutta tällöin robotin ohjelmaan tulee helposti turhia liikkeitä, jotka lisäävät läpimenoaika.

Omaa tuotantoa suunniteltaessa on siis hyvä muistaa vastuullisuus ja pitää loppuratkaisu mielessä. Kiinalaisessa Alibaba-verkkokaupparyityksen varastolla toimii muun muassa lajittelurobotteja, jotka toimivat ihmisten kanssa samassa tilassa. Ne eivät vaadi erillisiä turvaverkkoja, mutta nekin voivat toimia eri tavoin, jos niiden eteen tulee esteitä. Yhden tietyn kuljetuksen nopeus voi siis olla ehkä hitaampi kuin ihmisen, mutta yleisesti robotit ovat ihmistä paljon nopeampia, tehokkaampia, tarkempia ja turvallisempia. Näiden robottien käyttöönotto lajitteluun vähensi muun muassa



ruumiillisen työn määrää 70% ja ne pystyivät tunnistamaan 18 000 eri nimikettä tunnissa. Nämä ovat hyvä esimerkki tulevaisuuden älylogistiikasta, joka on kokenut valtavaa kehitystä Kiinan verkkokaupan kasvettua vuodesta 2012 vuoteen 2016 mennessä 1,3 biljoonasta 5,2 biljoonaan Yuaniin (n. 679 miljardia euroa). Koko logistiikkakaupan tulos on noussut 400 miljardiin Yuaniin (n. 52 miljardia euroa) ja keskiporto kiinalaisen verkkotilausten lukumäärä 23 kuljetukseen vuodessa per henkilö. Tämä kaikki edistää teknologian ja myös robotiikan kehitystä Kiinassa valtavasti. (Watch an army of robots efficiently sorting hundreds of parcels per hour 2017)

### 3 VOIMAOHJAUS

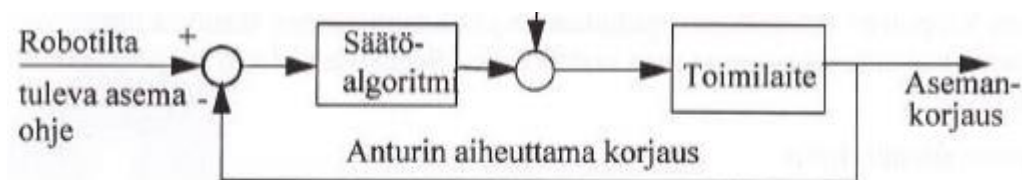
Voimaohjaus on yleisin sovellutus, mihin voiman- ja väännönmittausantureita käytetään robotiikassa. Voimaohjauksella tarkoitetaan sellaista ohjausta, joka tapahtuu mittaamalla voima-anturilla robotin tarkka sijainti työstövoimien perusteella. Voiman oikeanlaisella säätelyllä voidaan varmistaa, ettei robotti vahingoita itseään, ympäristöään tai lähellä olevia ihmisiä käyttämällä liikaa voimaa. Ohjaus on mahdollista voima-anturilla, joka mittaa työstövoimia eli liikettä kuudesta eri suunnasta, eli kuudella eri akselilla. Suunnat ovat kolme suorakulmaisen xyz-koordinaatiston pääsuuntaa sekä kiertosuunnat näiden ympäri. Yleensä anturi liitetään robotin käsivarren päähän ennen varsinaista robotin työvaiheessa käyttämää työkalua. Monissa antureissa on kaksi osaa, joista toinen on liikkumaton kotelo-osa ja toinen liikkuva painelevy. Kotelo-osa kiinnitetään robotin käsivarren päähän ja liikkuva osa työkalun päähän (Owen-Hill 2016a, 7-9, 17-18)

Tämän tyyppisellä ohjauksella robotin liikeratoja ei tarvitse ohjelmoinnissa määrittää täsmällisesti, vaan robotti kykenee anturin mittatietojen avulla kehittämään liikeratojaan pysyen samalla määritetyissä toleransseissa. Voimaohjaus kykenee tähän, sillä anturi havaitsee siihen vaikuttavat voimat ja vääntömomentit samanaikaisesti. Nämä havainnot antavat robotille ikään kuin tuntoaistin ja sijaintitiedon, jonka perusteella se voi kontrolloida robotin voiman käyttöä. Periaatteessa käyttäjä käskyy robotin suorittamaan tietyt voimat ympäristöön, minkä jälkeen robotti yrittää suorittaa nämä voimat. Jos robotti ei havaitse voimaa, se jatkaa automaattisesti navigoiden liikettään, kunnes haluttu voima on saavutettu. Ohjaus sisältää turvatoiminnon, joka voi tarvittaessa myös pysäyttää liikkeen, jos havaittu voima ei vastaa annettua käskyä ja käsketyin toiminnan jatkaminen voisi vahingoittaa joko robottia tai sen ympäristöä. Anturin keräämät mittatiedot välittyvät myös robotin työkaluun tai tarttujaan. (Erlbacher 2000, 2)

1970-luvulta lähtien tutkijat ovat etsineet keinoja siihen, miten roboteille voitaisiin lisätä tuntoaisti. Tästä huolimatta voimaohjauksesta on tullut yleistä vasta lähiaikoina. (Owen-Hill 2016b) Voimaohjauksen kehittyminen on mahdollistanut erilaisten haastavien pintakäsittelyn ja kokoonpanojen työvaiheiden automatisoinnit. Pintakäsittelyssä esimerkiksi robotin liikkeiden tai sen työkalun tulee myötäillä tai joustaa työstettävää pintaa vasten, jotta sitä voitaisiin käyttää viimeistelyihin. Tärkeää on jatkuva tasainen kosketus viimeisteltävään pintaan, että materiaalin poistettava tai lisättävä määrä pysyy tietyssä toleranssissa. (Erlbacher 2000, 1) Kokoonpanoissa taas asennettavien kappaleiden kulmat, asennot ja työntövoima ovat erittäin tärkeässä asemassa.

Voimaohjauksella mahdollistettu peg-in-hole -toiminto on yksiä parhaimpia esimerkkejä kokoonpano sovellutuksista. Nykyaikaiset kehitykset ovat mahdollistaneet yhä haastavammat kokoonpano sovellukset. (Owen-Hill 2016a, 18)

Robotti vertaa toiminnon suorituksen aikana ihmisen tekemän ohjelmoinnissa annettuja mittatietoja anturin sen hetkisiin tietoihin ja etsii poikkeamia näiden välillä. Mahdollisten poikkeamien löytyessä robotti ohjaa sen hetkisten tietojen perusteella tuntemansa mukaan liikettä automaattisesti oikeaan suuntaan. Tätä prosessia havainnollistaa kuvio 1. Työstövoimien mittausta mahdollistaa myös hyvin tarkkan liikepolun luomisen, joita voidaan vaatia ahtaissa työtiloissa. Voimaohjausta käytetään myös paljon hammaspyöräkokoonpanoissa. (Lempiäinen & Savolainen 2003, 139-142)



KUVIO 1. Voimaohjauksen säätökaavio (Lempiäinen & Savolainen 2003, 142)

Markkinoilla on tällä hetkellä tarjolla kahden tyyppisiä ratkaisuja voimaohjaukseen. Ensimmäinen näistä on adaptiivinen voimaohjaus, joka on näistä kahdesta kalliimpi toteuttaa. Adaptiivinen ohjaus on verrattavissa ihmiskäden toimintaan. Ajatus syntyi halusta yhdistää robotin ja ihmisen välinen analogia. Jos ihminen voi painamalla aiheuttaa työkaluun tietyn voiman, niin pystyy robottikin suorittamaan tämän saman voiman. Kuten ihmisillä on kädet ja lihakset, jota ohjaavat hermosto ja aivot, niin robotilla on käsi, jota ohjaa tietokone. Robottikäden päähän asennettava voima-anturi antaa tälle kädelle siis vielä ikään kuin tuntoaistin, joka lähettää mittatietoa suoraan robotin ohjaimelle. (Erlbacher 2000, 1-2)

Toisessa ohjaustavassa robotin päähän liitetään myötäilevä lisäakseli. Robottikäsi ja voiman aistiva työkalu ovat tässä tapauksessa erillisiä yksiköitä. Liitetyn lisäakselin työkalun liike voi olla joko lineaarinen tai kiertyvä. Työkalu tarvitsee yleensä oman ohjausyksikön, joka voi toimia yhteistyössä robottiohjaimen kanssa. Tämä toinen ohjausmenetelmä tarvitsee toimiakseen joustoa käyttämiinsä työkaluihin. Jousto työkaluissa varmistaa sen, että kosketus käsiteltävään pintaan ei muutu tai katkea toiminnon edetessä. Jouston saavuttamiseksi on olemassa voimansäätöohjauksia,

joita ovat passiivinen ja aktiivinen ohjaus. Passiivisen ja aktiivisen laitteen ero on voiman mittauksessa. Aktiiviseen työkaluun on sijoitettu voimaa mittaava komponentti. (Erlbacher 2000, 3-4)

Itselleen sopivaa vaihtoehtoa valitessa tulisi ottaa huomioon toimilaitteen mekaaninen maksimikuorma, jotta anturin tuoma turvallisuus on tarvittavalla tasolla. Tarpeen on myös miettiä sitä, mikä on anturin hinta, mitkä ovat vaaditut toiminnot ja applikaatiot, sekä paljonko aikaa integraatiolle on. Yleensä adaptiivinen ohjaus vaatii enemmän integrointia, ellei kyseessä ole tunnetulle robotille juuri erikseen yhteensopivaksi muotoiltu mittalaite. Esimerkiksi Robotiq -yrityksen valmistamat voima/vääntöanturit ovat yhteensopivia Universal Robots:in robottien kanssa. Jos anturi ei sisällä valmiiksi näitä yhteensopivuuksia, niin ne pitää koodata itse ja tähän voi kulua paljon aikaa. Yleensä myötäilevän lisäakselin käyttö on helpompaa, mutta se ei sovellu tilanteisiin, jossa halutaan mitata voimia monista eri työpisteen kohteista. (Owen-Hill 2016a, 9)

## 4 LAITTEET JA OHJELMISTOT

Tässä luvussa käydään läpi työhön tarvittavien laitteiden ja ohjelmistojen teoriaa. Teoria auttaa ymmärtämään, mitä ja millä laitteilla sekä ohjelmistoilla tässä työssä on tehty. Lisäksi teoria antaa kuvan siitä, mitä kaikkea näillä laitteilla ja ohjelmistoilla voidaan tehdä. Teoria auttoi myös sisäistämään paremmin näiden laitteiden ja ohjelmistojen ominaisuuksia, koska esimerkiksi ArtiMinds oli minulle ja Centrian henkilöstölle uusi ohjelma. Voima-anturin toiminta ei ollut myöskään tuttua käytännössä, vaan minulla oli siihen lähinnä vain teoriasta pintakosketus.

### 4.1 UR3-robotti

UR3 (KUVA 3.) on Universal Robots yhtiön uusi pienempi yhteistyörobotti, joka soveltuu pieniin kokoonpano- ja automaatioalustatehtäviin. Robotti painaa vain 11kg, sen maksimi työkuorma on 3kg ja toimintasäde on 500 mm. Jokainen nivel pyörii 360 astetta ja sen yläpään viimeinen nivel voi pyöriä akselinsa ympäri äärettömästi. Lisäksi sen joustavuus, keveys ja yhteistyökyky ovat erinomaiset. Tämä robotti on hyvä vaihtoehto niille, joilla on tarvetta 6-akselisille ominaisuuksille. Eritoten nämä sopivat sellaisiin sovellutuksiin, missä robotin koko, tietty turvallisuus taso ja hinta ovat kriittisiä valintakriteerejä. Universal Robots lupaa kaikkien robottiensa maksavan asiakkaan robottiin tehdyn investoinnin takaisin vain 195 päivässä, joten niihin tehty investointi maksaa itsensä takaisin nopeasti. Roboteilla on lisäksi TÜV-sertifikaatti. (Universal Robots 2019b)



KUVA 3. UR3-robotti (Universal Robots 2019)

UR3 sisältää patentoidun intuitiivisen 3D-visualisointiohjelmointiohjelman, jota voidaan käyttää liikuttamalla robotin käsivarsia tai koskettamalla tablettiohjaimen näytössä olevia nuolia. Enemmän kuin 80% Universal Robots:in myymistä roboteista työskentelee ihmisten kanssa ilman minkäänlaisia erillisiä turvaverkkoja, koska ne ovat yhteistyörobottien edelläkävijöitä turvallisuudessa. Robottia voidaan myös käyttää sellaisissa tiloissa, missä työntekijät eivät pysty normaalisti toimimaan, kuten vaarallisten tai myrkyllisten materiaalien läheisyydessä. Lisäksi robotit ovat hyviä vaihtoehtoja paljon toistoja sisältäviin tehtäviin, joita ihmiset eivät tee mielellään. UR3 on myös helppo tarvittaessa siirtää ja sijoittaa uudelleen uuteen tehtävään suorittamaan toimintoja, joten sillä voidaan automatisoida melkein mikä tahansa manuaalinen tehtävä, kuten esimerkiksi pienen tuote-erän valmistus. (Universal Robots 2019a)

Yleensä robotin laatikosta ulos ottamisesta käyttöönottoon vaatinut aika on alle tunnin. Tähän lukeutuu robotin kokoaminen sekä yksinkertaisen ohjelman teko. Varsinainen käyttöönotto robotin ensisijaisiin tehtäviin kestää yleensä noin puoli päivää lähetyksen purkamisesta. (Universal Robots 2019a)

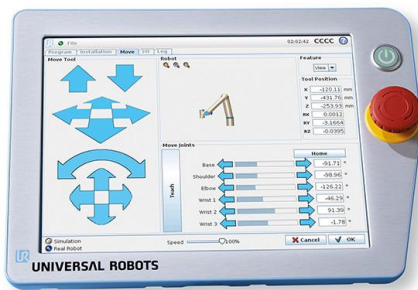
Robotin suunnittelussa on yritetty mukailla ihmisen käden toimintoja. Se saadaan toimimaan kuten ihmiskäsi ja jopa paremmin hyvällä ohjelmoinnilla. Yksi UR-robottien etuja on sen helppo ja intuitiivinen ohjelmointi, joka poistaa tarpeen kalliille kolmannen osapuolen toimijoille, joita yleensä tarvittaisiin aina, kun robotti haluttaisiin ohjelmoida uuteen tehtävään. Toisin sanoen vähennetään siis konsultoinnin tarvetta. Robotin ohjelma mahdollistaa sen, että kokematonkin ohjaaja saa nopeasti käsityksen perusohjelmoinnista. Robotilla voidaan asettaa reittipisteitä vain liikuttamalla sitä tarvittavaan asemaan. Toistuvia tehtäviä voidaan tallentaa UR-robotin muistiin, josta niitä voidaan käyttää aina uudestaan. (Universal Robots 2019a)

UR3 sisältää 15 erilaista säädettävää turva-asetusta. Yksi näistä on muun muassa ainutlaatuinen voimanmittausanturi, joka säätelee sitä voimaa, millä robotti voi törmätä työntekijään. Sen perusasetus pysäyttää kaiken liikkeen kohdatessaan 150 Newtonin voiman, mutta se voidaan säätää reagoimaan niinkin pieneen kuin 50 Newtonin voimaan. (Universal Robots 2015)

Robotilla pystyy muun muassa nostamaan ruuveja, asettamaan ne paikoilleen ja kiristämään tai löysäämään ne. Liimaamisessa robotti pystyy esimerkiksi annostelevaan saman tarkan määrän liimaa ja levittämään sen tarkasti. UR3 käyttää samaa 0,1 mm:n toistettavuustoleranssia, kuten muutkin UR-robotit. Se pystyy seuraamaan tarkkoja pintoja, kuten älypuhelimien alumiinireunaa. Robotti pystyy

tekemään tämän “tuntumalla”, eikä käyttämällä tarkkoja ohjelmoituja liikkeitä ja koordinaatteja. (Universal Robots 2015)

UR3-ohjauspaneeli on tabletin tyyppinen ohjain. Siihen on taakse kiinnitetty hihna parantamaan käyttäjän otetta ohjaimen putoamisen välttämiseksi. Tässä ohjaimessa on perinteisen vipukytkimen sijaan käytetty painiketta, jota painamalla robottia voidaan ohjata käsiohjauksella. Muuten tabletista löytyy hätäkatkaisin, virtapainike ja kosketusnäyttö, jossa sijaitsevat kaikki muut ohjaukseen ja ohjelmointiin tarvittavat objektit. Ohjain nähdään kuvassa 3. Ohjaimen mukana tulee yleensä myös kosketuskynä, joka toimii osoittimena kosketusnäytön käytön helpottamiseksi.



KUVA 3. UR3-robotin ohjain/ohjauspaneeli (ATN 2019)

Tässä työssä käytettävässä UR3-robotin käsivarteen on asennettuna OptoForcen HEX-70-XE-200N voima-anturi ja tarttujan päähän on asennettu Robotiq:n valmistama Robotiq 85, joka on mukautuva kaksisormitarttuja. Tarttujan omat kumiset päät on vaihdettu pois ja tilalle on asennettu muoviset 3D-tulostetut päät, joiden kärjessä olevat lisäosat ovat sormien kaltaiset. Kyseinen Robotiq:n tarttuja on eritoten tehty UR-roboteille, vaikka ne ovat yhteensopivia muidenkin markkinoilla olevien robottien kanssa. Tarttujan toimintoja voidaan käyttää ja muokata robotin ohjaimella. Ohjaimessa on tarttujalle oma valikko, josta pystyy muokkaamaan sen liikkeitä, nopeutta ja voimaa, jolla se tarttuu kappaleeseen. Lisäksi tarttuja voidaan säätää aukeamaan ja/tai sulkeutumaan tietyssä pisteessä.

## 4.2 ArtiMinds -ohjelmointiohjelma

ArtiMinds RPS (Robot Programming Suite) on Saksalaisen ArtiMinds Robotics:in tekemä ohjelmointiohjelma. Se mahdollistaa nopean ja intuitiivisen ohjelmoinnin ja simuloinnin

monimutkaisille robottitehtäville, mukaan lukien antureita vaativat tehtävät. Ohjelmointia ja simulointia voidaan suorittaa niin Online- kuin Offline-tilassa. Näiden tilojen pääpiirteinen ero on se, että Offline-tila käyttää RPS:n 3D-näkymää demonstraationa ja Online-tila oikean robotin asentoa. Vaihtoehtoisesti ArtiMindsia voidaan soveltaa liitettyllä anturilla ohjelmointiin käytettäessä ulkopuolista voima-anturia, kamera- tai monitorijärjestelmää. (ArtiMinds 2019) Ohjelmisto toimii normaalilla tietokoneella ja se on liitettynä robotin ohjaimen ohjelmoinnin aikana. ArtiMinds on vuorovaikutuksessa ohjaimen kanssa, jonka kautta se kerää prosessitietoja, kuten mittatietoja ja parametreja. Ohjelmistolla voidaan myös suorittaa liikkeitä tai hallinnoida ääri- ja lisärajoja, kuten tarttujan toimintaa. (Jäkel ym. 2016, 4)

ArtiMinds RPS Force Module on suunniteltu tekemään robotti älykkääksi. Voimaohjatuin liikkein robotti pystyy osoittamaan prosessin vaihtelut reaaliajassa ja mukailemaan liikkeitään kuten ihminen. Nämä liikkeet perustuvat voima-vääntömomentianturien lukemiin. Voimaohjauksella voi selvittää monet vaikeimmatkin automaation haasteet nopeasti. Näitä ovat esimerkiksi tarkkojen toleranssien asennukset, muoviosien asennukset, tiiviit pakkaukset, johdotukset, elektroniikka kokoonpanot tai kiillotukset. (ArtiMinds 2019)

Lista tuetuista robotti-, tarttija- ja voimanmittaustoiminnoista riippuu siitä, mitä versiota ohjelmistosta käyttää. Ohjelmoinnin aikana käyttäjä on suoraan yhteistyössä robotin kanssa opettaakseen sille uusia asentoja. Tämä tehdään tarvittavien pisteiden kautta joko ohjelman 3D-näkymästä tai liikuttamalla robottia manuaalisesti robotin omalla ohjaimella. Käyttäjä on yhteistyössä ArtiMinds RPS:n kanssa ohjelman rakenteen määrittämisessä ja erilaisten työkalujen valinnasta, joilla mahdollistetaan yksinkertainen parametrisointi. (Jäkel ym. 2016, 4)

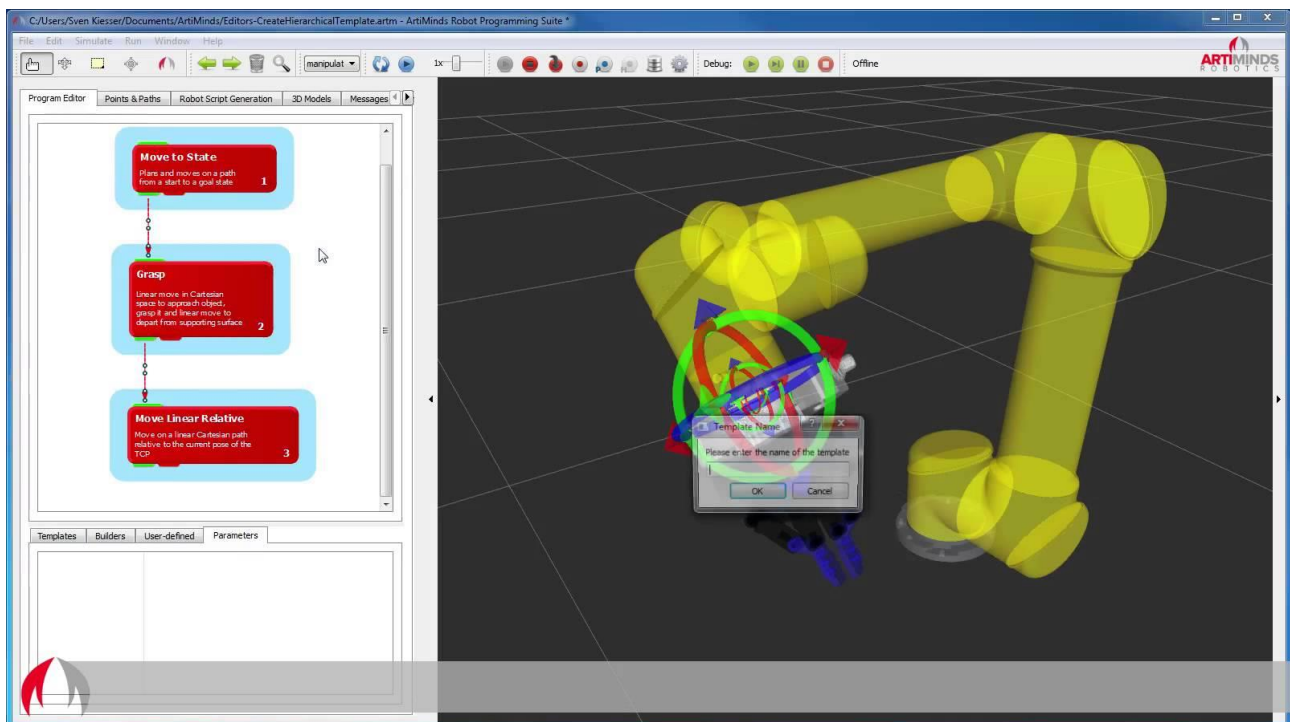
ArtiMinds RPS laskee, simuloi ja luo automaattisesti ymmärrettävän ohjelman, jota kutsutaan ohjelmakoodiksi. Tämän jälkeen koodi siirretään robotin ohjaimelle, jossa se pystytään suorittamaan itsenäisesti ilman PC:tä ja ArtiMinds RPS -ohjelmaa. PC ja ArtiMinds RPS voidaan aina tarvittaessa kytkeä takaisin kiinni ja tallennettu ohjelmakoodi on uudelleen ladattavissa, minkä jälkeen ohjelmaan voi tehdä muokkauksia. Esimerkiksi muuttaa robotin ohjaamista halutulla tavalla. (Jäkel ym. 2016, 4)

ArtiMinds-ohjelma koostuu erilaisista ryhmistä valmiita malleja. Jokainen malli sisältää sekä sisä- että ulostulot sekä eri signaaleja ja vastaanottimia, jotta sen peräkkäiset toiminnot onnistuvat



toivotulla tavalla. Aloittelija-asetuksella ainoastaan signaalit ja vastaanottimet ovat näkyvissä. Ainut vastaanotin on salli/aktivoi. Signaalit ovat kelpaava tai virhe. Kelpaava- tai virhesignaali mallissa A yhdistetään aktivoi-vastaanottimeen mallissa B. Tarkoituksena on, että B tulee suoritetuksi A:n jälkeen, kun A on suoritettu joko onnistuneesti tai epäonnistuneesti. (Jäkel ym. 2016, 10)

Ohjelman rakenne määräytyy sen eri jaksoista malleja, silmukoita ja hyvin sopeutuneita haaraumia. Täten käyttäjä valitsee joukosta ennalta- tai itsemääriteltyjä Template malleja ohjelman tietokannasta, joiden pohjalta tehdään haluttu kokonainen ohjelmakoodi. Yksi esimerkki valmiista ennaltamäärätyistä Template mallista on Grasp (tarttua). Tässä mallissa robotti avaa tarttujan halutussa pisteessä, minkä jälkeen se etenee määritettyyn pisteeseen tartuttavan objektin luo. Siinä se sulkee tarttujan ja nostaa objektin kyseiseltä pinnalta suorassa linjassa ulospäin siitä suunnasta, jossa robotti tarttui objektiin. Ohjelman parametrusointi toteutuu käyttäjän toimesta erinäisiä työkaluja käyttäen, joita voidaan yhdistellä keskenään. Näitä työkaluja ovat esimerkiksi Wizards, 3D editor ja Textual editor. (Jäkel ym. 2016, 10) Kuvassa 5 on nähtävissä ArtiMinds RPS:n Program Editor näkymä.



KUVA 5. ArtiMinds RPS:n Program Editor näkymä, missä on kolme Template mallia liitettyinä toisiinsa.

Wizard on graafinen, askel askeleelta asteittain ohjaava työkalu, joka johtaa käyttäjän mallin parametrisoinnin läpi. Siinä käyttäjää ohjataan joka vaiheessa tekemään jotain yhdestä seuraavista toimista:

1. Robottijärjestelmän ohjaaminen manuaalisesti johonkin tiettyyn asentoon. Esim. Asento, jossa robottijärjestelmä voi tarttua kohteeseen.
2. Manuaalinen yhden tai useamman numeraalisen parametrin syöttäminen. Esim. Robottijärjestelmän etäisyys verrattuna kohteeseen siinä kohdassa, jossa tarttumisprosessi voidaan aloittaa.
3. Anturidatan tallentaminen tiettyssä asennossa. Esim. Voima arvon mittaaminen.

3D Editor sallii käyttäjän hiirtä käyttämällä luoda ja muokata robotin asentoja kuin myös sen tallentamia pisteitä 3D-näkökulmasta. Textual editor -työkalulla mallin parametreja voidaan muokata taulukkonäkymässä kirjallisesti. (Jäkel ym. 2016, 11)

#### **4.2.1 Template -mallit**

Template-mallien yläpuolella näkyvät mallin sisääntulot ja vastaanottimet sekä vastaavasti ulostulot ja signaalit mallin alapuolella. Ne ovat korostettu eri värein. Esillä olevat pisteet merkitsevät suoritettun liikkeen keskeisimpiä paikkoja. Ne varmistavat nopean pääsyn käsittelyyn. Pieni valintaruutu osoittaa, jos liikeratoja täytyy laskea tässä mallissa. Jos se ei ole aktivoituna, aiemmin käytetyt liikeradat pysyvät sellaisenaan. Numerot oikeassa reunassa näyttävät sen, missä järjestyksessä mallit suoritetaan. Malli 1 suoritetaan ensin ja sitten Malli 2 jne. (Jäkel ym. 2016, 22-23)

Malleja on useanlaisia ja ne on jaoteltu eri kategorioihin niiden erojen ja niiden lopputavoitteen perusteella. Ohjelma parametrisoi malleja kahdella eri periaatteella, B: Basic ja F: Force. Basic -mallit käyttävät normaalia parametrisointia ja Force-mallit käyttävät voimaohjausta parametrisointiin. Malleista on myös sanalliset kuvaukset ja hiiren vieminen mallin päälle käynnistää lyhyen apuvideon demonstroimaan sitä, miten kyseinen malli toimii. On olemassa myös hierarkkisia malleja, jotka sisältävät itsessään useamman alamallin. Nämä mallit sopeutuvat varsinkin monimutkaisiin robotilla suoritettaviin tehtäviin. (Jäkel ym. 2016, 26)

Malleja on jaoteltu muun muassa seuraaviin ryhmiin. Move, Basics, Grasping, Path, Insert, Search, Align ja Storage. Nämä ryhmät sisältävät useita nimensä mukaisen tapaisia suoritemalleja. Näistä eri malleista muodostetaan itse valmiit ohjelmakoodit käyttäen uutta Drag & Drop -menetelmää, jossa valmiita malleja raahataan hiirellä Graph Editor -alustalle. Malli vedetään hiirellä haluttuun kohtaan ja vapautetaan siihen. Nyt uusi sijoitettu malli liittyy suoraan muihin jo ohjelmassa oleviin malleihin ja siihen vaikuttaa aina kaksi mallia, edellinen ja seuraava. Malleja voi lisätä myös ohjelmaan valitsemalla yhden tai useamman malleista ja käyttää Copy & Paste -toimintoa, jolla voidaan kopioida tietty malli tai vain tietty osa ohjelmaa. Liitettäessä kopioitu osa liittyy suoraan uuden ohjelman muihin malleihin. Automaattiset liitännät ohjelmassa saattavat kuitenkin aiheuttaa ei-toivottuja lopputuloksia, jotka voidaan palauttaa takaisin ennen liittämistä vallinneeseen tilaan käyttämällä Undo -toimintoa. Mallien liittymisjärjestystä voidaan muuttaa missä vaiheessa vain. Undo -toimintoa voidaan myös käyttää yksitellen vain haluttuihin malleihin kokonaisuuden sijaan. Redo -toiminnolla saadaan palautettua se tila, joka vallitsi ennen Undo -toimintoa. Ohjelmakoodi voidaan aina tallentaa ja ladata uudelleen käyttäen perinteisiä Save-, Save As- ja Open File -komentoja. (Jäkel ym. 2016, 26-58)

#### 4.2.2 Wizards -apuohjelmat

Wizards ovat graafisia havainnollistavia apuohjelmia, jotka etenevät askelaskeleelta eteenpäin auttaen Template-mallia kohteen parametrisoinnissa. Käyttäjä ohjataan vähitellen parametrisoinnin läpi osoittamalla joukko Key Points-avainpisteitä käyttäen joko oikean tai 3D-näkymän robotin asemaa. Se, kummasta demonstraatiosta ohjelma tallentaa mittatiedot, on riippuvainen siitä, käytetäänkö ohjelmaa Online- vai Offline-tilassa. Online-tilassa ohjelma käyttää robotin oikeaa asemaa ja Offline-tila käyttää ohjelman 3D-näkymän robotin asemaa. Offline-tilalla ohjelmoidessa on hyvä sisältää ohjelmaan 3D-malli työpisteestä, jotta robotti ymmärtäisi ympäristönsä paremmin. Avainpisteet siirtyvät ja tallentuvat ohjelmalle joko ulkoisista antureista, kuten esimerkiksi voima-anturista, tai sisäisistä antureista, kuten asema-anturista. Näitä avainpisteitä käyttäen ohjelma laskee itse automaattisesti mallin kaikki parametrit ja simuloi mahdollisimman optimaalisen toiminnon. Jokaisessa mallissa on vähintään yksi Wizard. (Jäkel ym. 2016, 58)

Esimerkiksi voimaohjausta käyttävä toiminto Spiral Search Contact -mallissa oleva Wizard käy läpi spiraalin tavoin eri avainpisteet. Näitä ovat esimerkiksi ylä-vasen-kulma, ala-oikea-kulma ja keskipiste. Napauttamalla kohtaa Use Demonstration ohjelma tallentaa robotin parametrit ja voimat

sen asennon mukaan. Spiral Search Contact määrittelee myös mahdollisiin aukkoihin sen minimi- ja maksimisyvyyden, johon asti robottikäden on liikuttava, jotta tarvittava toiminto suoritetaan onnistuneesti. Tarvittaessa parametreja voidaan antaa myös numeerisesti. (Jäkel ym. 2016, 60-70)

### **4.2.3 Visualisointi ja simulointi**

Ennen kuin oikealla robotilla voidaan suorittaa ohjelman toimintoja, robotin ohjelmakoodi on simuloitava. Kaikkia simuloituja liikkeitä voidaan tarkastella ohjelman 3D-näkymässä. Ohjelmakoodi latautuu ohjelmalle tarkasteltavaksi painettaessa simulointi painiketta. Annettujen tietojen pohjalta ArtiMinds laskee ja luo optimaaliset kulkureitit itsenäisesti parametrien puitteissa. Tämän jälkeen painetaan visualisointi-painiketta, jotta voidaan esimerkiksi tarkistaa, tuliko ohjelmakoodiin törmäyksiä. Robotti ei välttämättä näe kaikkia niitä esteitä, joita oikeassa suoritustilanteessa on. Tästä syystä on tärkeää yrittää verrata tilannetta mahdollisimman paljon oikeaan, jotta kaikki mahdolliset vaaratilanteet voitaisiin välttää. (Jäkel ym. 2016, 72-76) Tähän auttaa oikeaa työaluetta vastaavan 3D-mallin lisääminen ohjelmaan.

Visualisoinnin 3D-näkymä korostaa aina sillä hetkellä simuloinnissa tapahtuvaa liikettä Program Editor-ikkunassa viivamaisesti. Kaikki liikkeet ovat myös värikoodattuja. Väreinä toimivat vihreä, keltainen ja punainen. Vihreä viiva tarkoittaa nopeaa liikettä ja punainen hidasta. Värien vaihtelu ilmoittaa prosessissa tapahtuvasta kiihtymisestä. (Jäkel ym. 2016, 76)

Ohjelmassa voi myös tarvittaessa visualisoida tietyn mallin kaikki eri pisteet valikon Points & Paths alta. Mallin eri välivaiheet on ilmaistu mallin yläkulmassa sijaitsevilla pisteillä. Nämä pisteet ovat näkyvillä vain ensimmäisen simuloinnin jälkeen. Painamalla eri pisteitä 3D-näkymä päivittyy näyttämään ne kohdat, joissa robotti sijaitsee näillä eri välietapeilla. Kaikista oikeanpuolimaisiin näyttää kyseisen mallin päätepisteen ja vasemmanpuolimaisiin aloituspisteen. (Jäkel ym. 2016, 78-80)

### **4.2.4 Esimerkki ohjelmakoodin luomiseen**

Seuraavaksi esimerkki siitä, miten perus ohjelmakoodi luodaan. Ei ole väliä mihin sovellukseen ohjelma tulee tai mitä Template-malleja käyttää. Periaate ohjelmoinnissa on kuitenkin aina sama.

1. Valitse Template-malli mallikirjastosta. Jotta sopiva malli löytyisi nopeasti ja tehokkaasti, hakukenttä antaa mahdollisuuden käyttää hakusanoja tai kuvauksia mahdollisista tarvittavista toiminnoista.
2. Lisää haluttu Template-malli robotin ohjelmaan Drag & Drop -menetelmällä ja määrittele mallille toimintajaksot.
3. Avaa mallin Wizard antaaksesi parametrit uudelle mallille.
4. Seuraa askeleittain ohjelman antamia ohjeita ja liikuta robottia antaaksesi sille Key Points -avainpisteet joiden avulla ohjelma asettaa sille parametrit.
5. ArtiMindsin simuloinnin ja liikkeiden luonnin automaattiset laskelmoinnit yhdistävät mallien toiminnot toisiinsa. Käyttäjän tulee varmistaa visuaalisesti 3D-näkymästä simulointien avulla, että mallit ja niiden toiminnot ovat yhdistyneet halutulla tavalla.
6. Tarvittaessa tarkistetaan ja hienosäädetään kiertoarvojen jälkimuokkaus, nopeus, kiihtyvyyys ja yhdistely. Käyttäjä varmistaa jälleen lopputuloksen 3D-näkymästä.
7. Ohjelmakoodi siirretään lähettämällä se ohjelmalta robotin omalle ohjaimelle. Siirrettäessä ohjelma automaattisesti kokoaa ohjelmakoodin, joka siirtyy robotille sen omalla alkuperäiskielellä.
8. Koottu ohjelmakoodi voidaan nyt suorittaa robotin omalla ohjaimella. Ohjelmiston tai tietokoneen ei tarvitse enää olla liitettynä robottiin, vaan ne voidaan irrottaa. (Jäkel, ym. 2016, 14-22)

### **4.3 OptoForce voima-anturi**

OptoForcen 6-akselinen HEX-70-XE-200N High Precision voiman/väännönmittausanturi (KUVA 6) mahdollistaa voiman ja väännön mittaamisen kuudella eri asteella. Anturi kykenee siis mittaamaan jokaiseen suuntaan etenevää liikettä, mikä mahdollistaa tarkan voimaohjauksen. Kuuden suunnan mittaaminen mahdollistaa hyvin tarkat, vaativat ja monipuoliset asennukset. Anturi on yhteensopiva

suureen osaan teollisuustason roboteista. Yleisiä käyttökohteita on voimankontrollointi ja voimaohjaustehtävät, imitoimalla oppivat tehtävät sekä törmäysten havaitseminen. Antureita käytetään myös paljon lopputuloksen parantamiseen esimerkiksi hionnassa, kiillotuksessa ja purseenpoistossa. Anturin mitat ovat korkeus - 35mm x halkaisija - 75mm. Anturi tukee USB, CAN, EtherCAT ja Ethernet -TCP/UDP liitäntöjä. Tuettuja alustoja ovat Windows, Linux, ROS ja UR. (OptoForce HEX-70-XE-200N pdf) Tekniset tiedot ovat nähtävissä taulukossa 2.



KUVA 6. OptoForce 6-Axis Voima-anturi (IFR 2017)

TAULUKKO 2. OptoForce HEX-70-200N tekniset tiedot

<b>OptoForce HEX-70-XE-200N</b>		
<b>Tietoja</b>	<b>Etuja</b>	<b>UR sovellukset (URaps)</b>
Puristus (Fz) 200N	Moniakselinen voimanmittaus	Läsnäolotunnistin (Presence Detection)
Veto (Fz) 200N	Korkea resoluutio	Keskipiste paikannus (Centerpointing)
Kapasiteetti (Fx, Fy) 200N	Erittäin mukautuva tuotteen suunnittelu	Käsinohjaus (Handguiding)
Vääntö (Tz) 6,5Nm (Txy) 10Nm	Pöly- ja vesisuojattu (IP65)	Polun tallennus (Path Recording)
	Suuri ylikuormitus alue	Kiillotus (Polishing), muovi ja metalli
	Mekaanisesti iskunkestävä	Asennus (Box Insertion)
	Kustannustehokas ratkaisu	Pujotus (Pin in the hole)
	Helppo integrointi	Pinoaminen / Poistaminen (Stacking/Destacking)
		Palletointi (Palletizing)
		Hionta (Sanding)

#### 4.4 SolidWorks 3D CAD -suunnitteluohjelma

SolidWorks on ranskalaisen Dassault Systèmes -yrityksen valmistama parametrinen 3D-mekaniikkasuunnitteluohjelmisto, joka sisältää myös tilavuus- ja pintamallinnustyökalut. Ohjelmistoa käytetään hyvin erilaisiin tarkoituksiin, kuten koneiden, laitteiden ja yksittäisten osien tai kappaleiden suunnitteluun. SolidWorksin ensimmäinen versio julkaistiin vuonna 1995 ja uusia versioita tulee vuosittain. Kaikki versiot ovat Windows-pohjaisia, mutta jotkut uudemmat versiot eivät toimi vanhoissa Windows -järjestelmissä. Myöskään SolidWorksin versiot eivät ole alaspäin yhteensopivia eli uudemmalla SolidWorks versiolla tehdyt mallit eivät toimi vanhemmissa versioissa. Vanhat työt ovat kuitenkin ylöspäin yhteensopivia eli ne toimivat uudemmissa versioissa. Tässä projektityössä käytettiin vuoden SolidWorks 2017 -versiota. (SolidWorks Wikipedia 2019)

Ohjelmistolla voidaan tehdä kolmenlaisia perustiedostoja: osia, kokoonpanoja ja valmistuspiirustuksia. Nämä ovat kaikki yhteydessä toisiinsa siten, että osa voidaan liittää kokoonpanoon tai osasta voidaan tehdä piirustus. Lisäksi ohjelmisto muuttaa osan mittoja sekä kokoonpanossa että piirustuksessa, jos käyttäjä muuttaa kyseisen osan sketsin mittoja. Samasta osasta pystytään myös tekemään monta eri versiota, minkä tarkoitus on helpottaa suunnittelijan työtä. (SolidWorks Wikipedia 2019)

SolidWorksin mallintaminen perustuu piirteisiin, jonka suurimpana erona 2D CAD -ohjelmistoihin on se, että käyttäjän luoma esine, osa tai kokonaisuus on muokattavissa erittäin vapaasti ilman että työ on aloitettava uudelleen alusta. Mallit yleensä valmistetaan ensin piirtämällä 2D-tasolle Sketch, minkä jälkeen sille voidaan tehdä haluttu toiminto. Tämä toiminto voi olla esimerkiksi ehkä yleisimmin käytetty Pursota (Extruded Boss), joka antaa Sketch-luonnokselle kolmannen suunnan tehden siitä kolmiulotteisen. Malleja voidaan tehdä myös monilla eri tavoilla käyttäen eri toimintoja, mutta kaikille näille on yleistä se, että yhtä ja ainutta oikeaa tapaa ei ole. Kaksi erilaista kokoonpanojen suunnittelutapaa on kuitenkin selvästi eroteltavissa: ns. bottom-up ja top-down. Nimensä mukaisesti ensimmäisessä valmistetaan ensin laitteen osat ja näistä tehdään isompi kokonaisuus eli kokoonpano. Jälkimmäisessä mallinnetaan uusia osia jopa täysin tyhjiin kokoonpanoon. Menetelmää voi vaihtaa työn edetessä. (SolidWorks Wikipedia 2019)

SolidWorksin ohjelmistoa myydään kolmea eri mallia, jotka ovat Standard, Professional ja Premium. Lisäksi ohjelmistoa voidaan laajentaa erikseen hankittavilla lisäosilla, jotka laajentavat ohjelmiston käytettävyyttä. Näillä saadaan lisätoimintoja, joita voi olla esimerkiksi erilaisten materiaalien lisääminen osaan. Lisäosien avulla voidaan esimerkiksi arvioida kappaleiden kestävyyttä sekä käyttäytymistä valmiina tuotteena. Ohjelma pystyy näin huomaamaan jo suunnitteluvaiheessa, että valmis tuote ei käyttäytyisi halutulla tavalla. Jotkin monimutkaiset toimintojen automatisoinnit vaativat yleensä lisäosia, mutta kevyemmät tehtävät onnistuvat luontevasti ilman lisäosia mainiosti. Ohjelmisto on hyvin käyttäjäystävällinen ja sen piirroksia voidaan linkittää muihin ohjelmistoihin, kuten esimerkiksi Microsoft Exceliin ja ArtiMinds RPS:ään. (SolidWorks Wikipedia 2019)



## 5 TYÖVAIHEET / TYÖN ETENEMINEN

Aloitin opinnäytetyön tekemällä työstä projektisuunnitelman. Siitä selvisi, mitkä ovat tämän työn tavoitteet, mitä on tarkoitus tehdä, miksi näin tehdään ja miten se aikataulutetaan. Suunnitelmassa oli myös mukana työn riskien arviointi, työn resurssit sekä suunnitelma raportoinnista ja seurannasta. Työvaiheet jaottelin Excel-taulukkoon, jossa oli kiinnitettynä eri tavoitteet eri aikamääreille. Tein myös Excelillä SWOT-analyysin työn riskeistä ja mahdollisuuksista, minkä liitin projektisuunnitelmaan. Suunnitelma esitettiin myös julkisesti opiskelijaryhmälle.

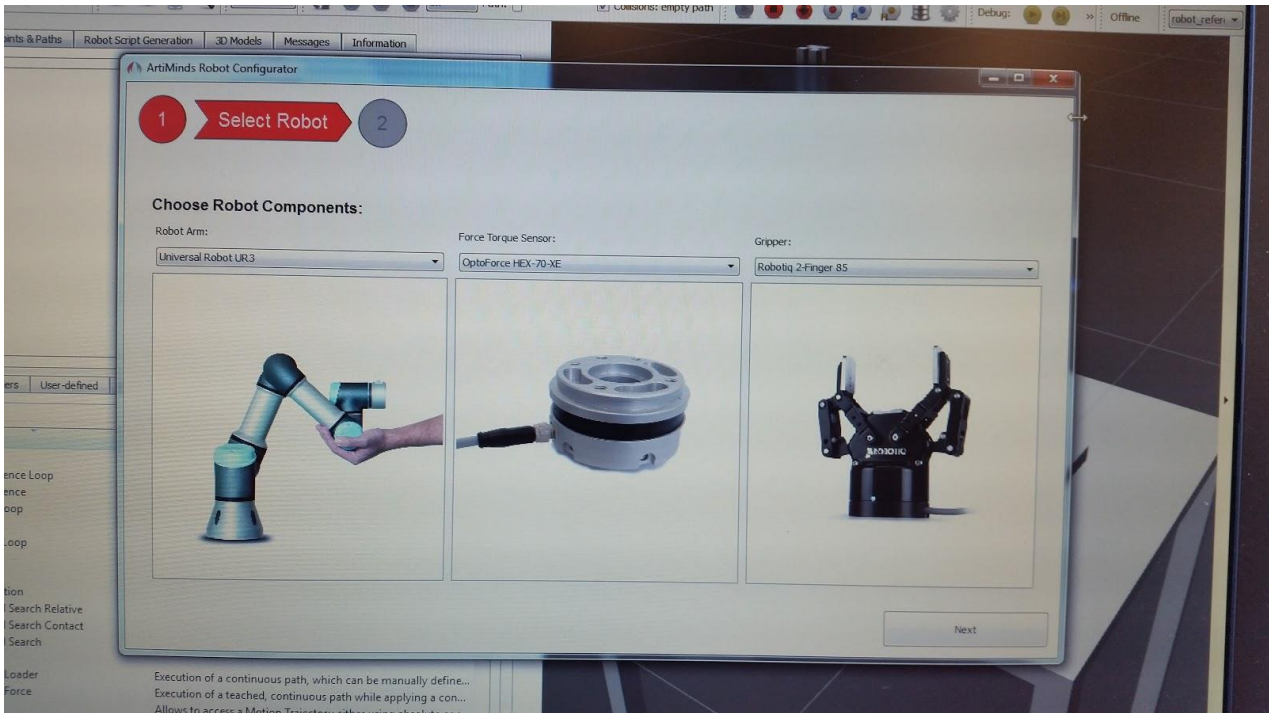
Suunnitelman esittämisen jälkeen alkoi teoriaan tutustuminen. UR3 oli itselleni tuttu aiemmista opinnoista, mutta siitä huolimatta perehdyin sen tietoihin ja teknillisiin ominaisuuksiin vielä tarkemmin. Tutustuin Universal Robotsin internetsivuihin, UR3-tuotesittelyyn, tuotteen mainoslehteen ja yrityksen omaan YouTube-kanavaan. Lisäksi tein heidän internetsivuillaan olevan 89 minuutin robotiikan koulutuksen. ArtiMinds puolestaan oli minulle kokonaan uusi ohjelmisto ja aloitin siihen tutustumisen käyden läpi heidän internetsivujaan, YouTube -kanavaa sekä lukemalla “ArtiMinds RPS Expert Training” -opasta. OptoForcfen voima/vääntö-anturiin sain teorian tiedot opettajalta sähköpostitse. Näistä kävi ilmi sen toiminnot sekä soveltuvuudet UR3:n kanssa.

Teoriaan tutustumisen jälkeen aloin kirjoittamaan kaikista näistä kolmesta aiheesta ensin teoriatietosiota, joka on esitetty tässä raportissa luvussa 4. Osat on tehty siten, että tätä raporttia lukeva saisi tarkemman käsityksen siitä, mitä eri ohjelmistoja ja laitteita työssä käytetään ja miten ne toimivat. Samalla kokosin itselleni tarvittavia perustietoja varsinkin ArtiMinds -ohjelmistoon, koska en ollut koskaan ennen edes kuullut siitä. Teoria myös auttaa mielestäni asioiden käytännöntoteutuksessa. On hyväksi olla asiasta pohjatietoa, niin mahdolliset ongelmatilanteet ovat helpompi ratkaista. Teorian ja käytännön kokemuksen kautta voi taas syventää oppeja konkreettiseksi tiedoksi.

### 5.1 Palletointi

Aloin tehdä ohjelmakoodia ArtiMinds RPS:llä uudelle tyhjälle alustalle aluksi ilman, että CAD-malli oli siihen liitetty. Ensiksi oli koottava ohjelman asetuksista oikeanlainen robottikonaisuus, jotta ohjelma ymmärtäisi robotin liikkeitä ja muitakin parametreja mahdollisimman hyvin. Tämä löytyi Configure-valikon alta kohdasta Custom Robot. Ohjelman robottiin liitettiin oikeanlainen tarttuja,

sekä OptoForcen voima-anturi. Tämä Configure on nähtävissä kuvassa 7. Ohjelmalle oli syötettävä oikeat IP-osoitteet ja tehtävä tarvittavat yhteystestit, jotta robotin ja ohjelman välinen kommunikointi toimisi halutulla tavalla. Toimiva yhteys mahdollistaa Online-tilassa ohjelmoinnin, jota tarvittiin ohjelman kykyyn tunnistaa oikean robotin asento annetusta demonstraatiosta.



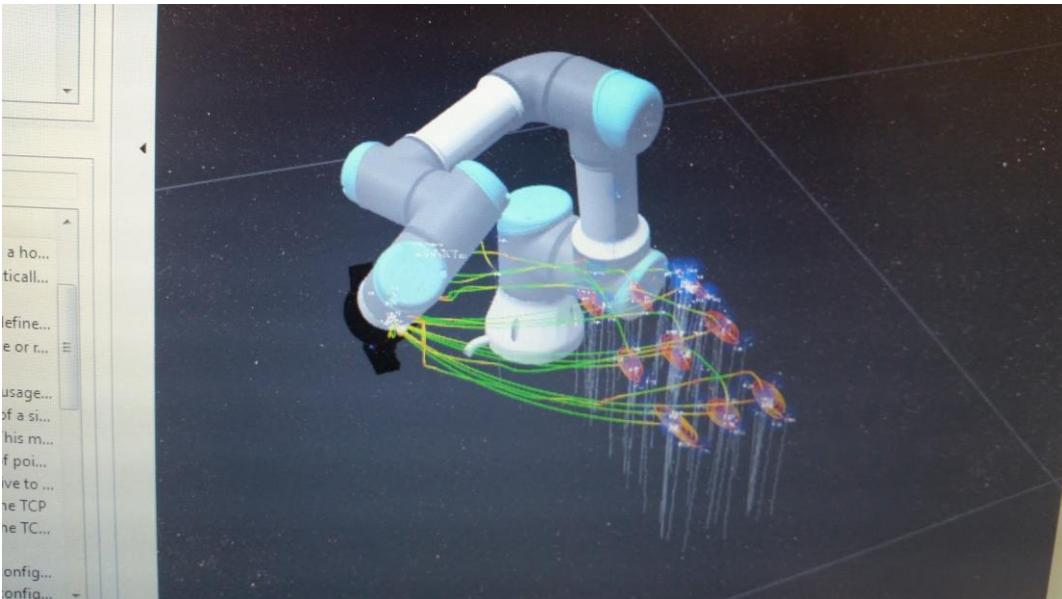
KUVA 7. Robotin kokoaminen Configure kohdassa. Kuvassa näkyy kaikki oikeat komponentit valittuna ohjelman robottiin.

Tämän jälkeen aloin ensiksi kokeilemaan, miten hyvin ja helposti ArtiMindsin valmiilla Template malleilla voi tehdä ohjelmakoodin. Käytin erilaisia malleja, jotka yhdistin järkeväksi kokonaisuudeksi. Kokeilin muun muassa, miten Spiral Search -voimaohjausta käyttävä toiminto toimii ilman ohjelmaan liitettyä 3D-mallia. Ohjelman teko onnistui pienen uuden ohjelman alkukankeuden jälkeen yllättävän helposti. Tein ohjelman pallein käyttäen Grid-mallia. Palletti on ruudukkomainen alue, missä on eri määrä ruutuja, joissa jokaisessa käydään tekemässä jokin tietty tehtävä. Tässä tapauksessa kyse oli pattereiden lajittelusta 3x3 kokoiselle ruudukolle, jossa jokaisessa oli pattereille tarkoitettu paikka, joka oli patteria hieman isompi kolo. Robotti osasi laskea ja luoda oikeanlaisen pallein, kun sille antoi ruutujen määrän sekä osoitti reunapisteiden sijainnin. Ohjelman käyttämät Wizardit tekevät ohjelmakoodin tekemisestä suhteellisen yksinkertaista. Wizardit ohjasivat haluttujen toimintojen parametrisoinnin läpi erilaisten avainpisteiden kautta, jotka tallentuivat

ohjelmalle joko syöttämällä sille manuaalisesti lukuja tai demonstroimalla haluttu paikka robotilla. Robottia voi ohjelman avulla siis periaatteessa opettaa imitoimalla.

Ohjelmakoodin simuloinnit toimivat ohjelmalla moitteettomasti, mutta huomasin myös valmiiden Template-mallien erot selkeästi. Aluksi käyttämäni Path-toiminto ei ollut optimaalinen käytettäväksi UR3-robotin käsiohjauksen kanssa. Robottia on muuten kyllä todella helppo ohjelmoida käsiohjauksella, mutta tähän toimintoon se ei sovellu ainakaan tässä yhteydessä. Tämä johtuu siitä, että jos käsiohjauksen aikana tapahtuu yksikään ylimääräinen liike, robotti tallentaa sen liikkeen ohjelman koodiin. Ohjelma siis tallensi liikkeen juuri sellaisena kuin se oli demonstroitu, eikä se tässä tapauksessa käyttänyt omia laskelmiaan liikeradan optimoimiseksi. Tämä johtaisi tuotannossa jatkuvaan robotin turhaan ja ylimääräiseen työhön, joka alkaisi vuoden mittakaavalla vaikuttamaan erittäin suuresti tuotannon määrään. Voisin kuvitella Path-toimintoa käytettävän vain ainoastaan sellaisissa tilanteissa, missä robotin ympärillä on paljon esteitä ja liikeratojen on oltava täsmällisiä. Toiminnot, kuten Move to State ovat paljon parempia tähän palletoinnin ohjelmakoodiin, koska tällöin ArtiMinds itse autonomisesti laskee ja valitsee suorimman mahdollisen ja toimivan liikeradan. Move to State mahdollistaa myös yksinkertaisen Loop-mallin käytön, koska se laskee automaattisesti parhaan reitin toistuville toiminnoille. Loop on siis malli, joka toistaa tietyt halutut toiminnot, ennen kuin se siirtyy seuraavaan mallin ohjelmakoodissa. Tämä ei toiminut aivan täydellisesti, koska ohjelmaan ei ollut liitettyä 3D-mallia työpisteestä.

Tekemässäni ohjelmakoodissa ArtiMinds laskee kyllä todella suorat liikeradat, mutta se ei aina välttämättä ymmärtänyt tarkasti ympäristön parametreja ja esteitä. Tuloksena robotin tarttumat osuivat puheen paletin reunaan Grid Loopin Move to State -toimintojen viidennellä liikkeellä. Tämä johti robotin turvajärjestelmän sammuttamaan liikkeen välittömästi ja löivät robotin lukittavat jarrut päälle, koska robotin voima-anturi tunnisti odottamattoman työstövoiman tapahtuneen. Robotin ohjaimelle tuli myös virheilmoitus siitä, että tässä tapauksessa se oli robotin "ranne", joka havaitsi törmäyksen ja siksi liike katkesi. Liikeratojen törmäysvapaaseen sekä tarkkaan ohjelmointiin ja simulointiin tarvitaan siis RPS:lle 3D-malli. Kuvasta 8 on nähtävissä 3D-näkymä palletoinnista ilman 3D-mallia.



KUVA 8. ArtiMinds RPS:n simulaation 3D näkymä palletoinnista ilman 3D-mallia. Näkyvissä myös eri liikkeiden viivamaiset värikoodit.

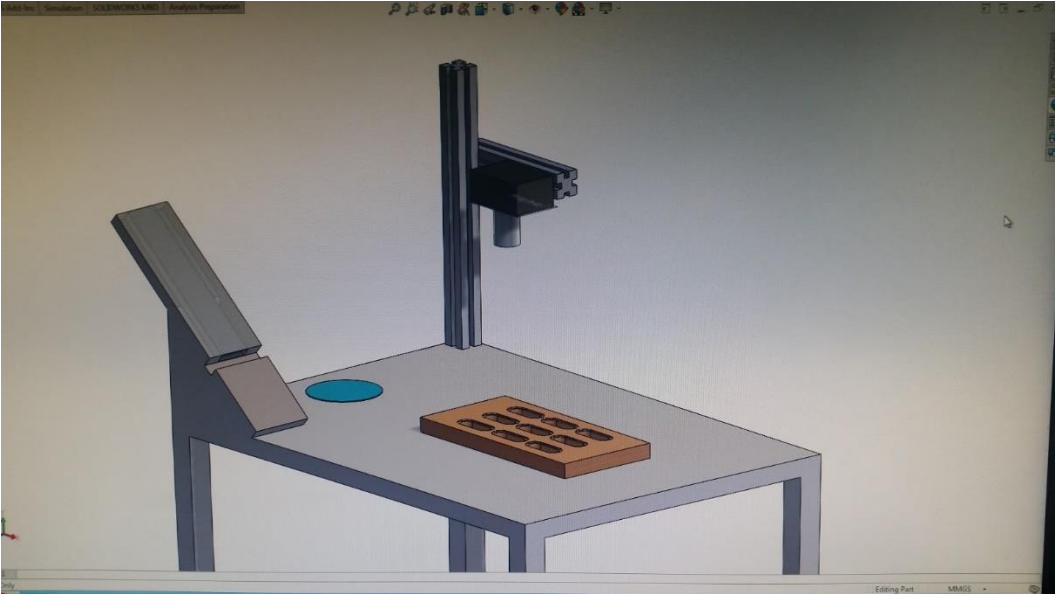
Tämän tehdäkseen ohjelman asetuksista on tarkistettava, että Beginner-taso ei ole päällä, koska vain Advanced- ja Expert-tasot mahdollistavat 3D-mallien lisäykset ohjelmaan. Beginner-tasolla CAD-mallista pystyy vain ottamaan kaarevia liikeratoja, mutta se ei sisällytä itse mallia RPS:lle. 3D-mallin kanssa ohjelma osaa huomioida laskea työpisteellä olevia esteitä, kunhan Path Planning -asetuksista on valittu Force Collision Checking päälle. Move Collision -mallia on hyvä käyttää silloin, kun on jo tiedossa, että robotin kulku-uralla on esteitä. Mallin Wizard opettaa esteen parametrit osoittamalla vähintään kaksi eri nurkkapistettä, jotka ohjelma muuttaa kuution malliseksi alueeksi, minkä se sisältää esteeksi. Toimintojen liikeradat siis kiertävät tai väistävät määritetyn kuution alueen mahdollisimman suoraa reittiä haluttuun loppupisteeseen.

Teoriaosien kirjoittaminen alkoi olla valmista myös tässä vaiheessa, joten aloitin työssä tarvittavan 3D CAD-mallin tekemisen. Menin työpisteelle (KUVA 9) ottamaan mittanauhalla oikeat mitat työpöydästä sekä kaikista työpisteellä olevista asioista, kuten patteritelineen, puupalletin ja konenäkökameran. Mittasin niiden etäisyydet toisistaan sekä niiden omat mitat ja kirjoitin ne ylös.



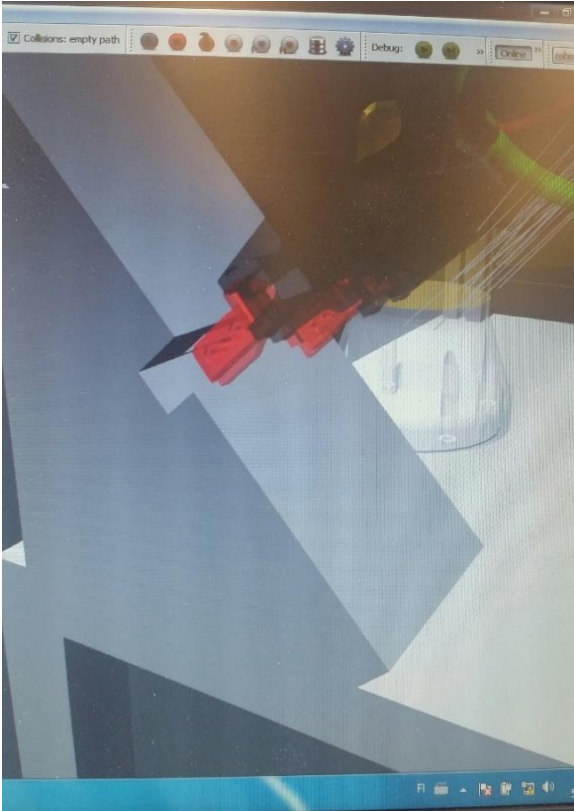
KUVA 9. Kuva varsinaisesta työpisteestä. Kuvassa näkyy myös UR3, OptoForcen voima-anturi, sekä Robotiq 85 2-Finger gripper asennettuna paikalleen

Tämän jälkeen menin tietokoneelle ja aloin piirtämään työpisteen mallia käyttäen SolidWorks 2017 CAD -suunnitteluohjelmaa. Malli valmistui nopeasti, koska SolidWorks oli minulle entuudestaan tuttu ohjelma. Piirsin ensiksi pöydän jalkoineen, jonka päällä kaikki ovat. Tämän jälkeen mallinsin pöydälle puupaletin, patteritelineen ja konenäkökameran. Merkitsin vielä lopuksi ympyrällä pöydälle robotin tarkan paikan. CAD-malli on nähtävissä kuvassa 10, jossa patteriteline, puupaletti, konenäkökamera ja robotin paikka ovat kaikki selkeästi nähtävissä eri väreissä. Malli muuttui tästä kuvasta lopulliseen työhön siten, että patteriteline tuli edemmäs.



KUVA 10. Näkymä SolidWorksillä tehdystä työpisteen 3D CAD-mallista. Robotin paikka on merkattuna sinisellä ympyrällä.

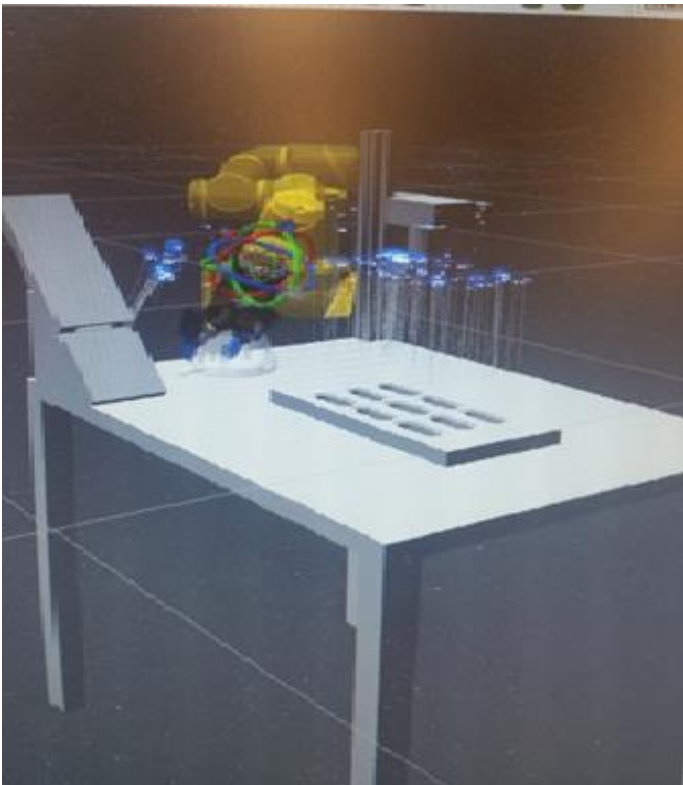
CAD-mallin valmistuttua palasin takaisin ArtiMindsin RPS:lle ja lisäsin mallin ohjelmaan. Malli siirtyi ohjelmaan aluksi nurinkurisessa asennossa, joten se täytyi ensin kääntää, oikaista ja liikuttaa oikealle paikalleen. Tätä helpotti työpisteen pöytään valmiiksi merkitty robotin tarkka paikka. Tässä vaiheessa kannattaa myös pitää valintaruutu Collisions valittuna, koska ohjelma ilmoittaa tällöin robotin törmäyksistä punaisella värillä. Tällöin robotti ei jäisi vahingossa leijumaan ilmaan, eikä menisi työpisteen sisään, vaan asettuisi juuri oikealla tavalla pöydän pinnalle. Esimerkki Collisions-toiminnon törmäyksen havainnoinnin demonstraatiosta on nähtävissä kuvassa 11.



KUVA 11. Collisions havainnointi 3D näkymässä, jossa sininen väri muuttunut punaiseksi merkiten törmäystä. Tässä tapauksessa kaksisormitarttujan osat olisivat osumassa patteritelineeseen.

Kokeilin ohjelmakoodin tekoa nyt myös Offline-tilassa 3D-mallin liitettynä. Robotille näytettiin sen tarvitsemat pisteet samalla tavalla kuin Online-tilassa, mutta nyt robottia liikuteltiin vain 3D-näkymässä. Huomasin, että ohjelman robotissa ei ollutkaan tismalleen samanlainen Robotiq 85 -kaksisormitarttuja. Todellisessa robotissa oli erityiset 3D-tulostetut muoviset päät, jotka kapenevat lopusta ohuemmaksi, jotta sen “sormet” yltaisivät patteritelineen syötölle. Näin ollen piti huomioida se, että 3D-näkymän tarkat paikat eivät saisi ulottua liian pitkälle, tai muuten robotin tarttuja osuisi palettiin. Ohjelmakoodin teko oli näin myös helppoa, mutta hitaampaa kuin Online-tilassa, sillä robottia piti liikuttaa hiirellä 3D-näkymässä olevista nuolista. Nuolista liikuttamisessa oli kuitenkin se hyvä puoli, että se piti liikuttaessa tarttujan samassa asennossa. Huono puoli oli se, että joskus tämä kuitenkin aiheutti muutamaan robotin niveleen sellaisia liikkeitä, jotka eivät olleet optimaalisia. Tätä tapahtui muutaman kerran, kun ryhdyin liikuttamaan robottia 3D-kuvassa. Robotin käsivarsi nousi ylös erikoisessa kulmassa, mikä aiheutti outoja, ylimääräisiä ja pitkiä liikkeitä, jotta robotti pystyi siirtymään seuraavalle kololle paletissa. Tämä toi ohjelmakoodiin paljon ylimääräistä työtä, joten se ei olisi optimaalinen.

Seuraavaksi kokeilin ohjelmakoodin tekoa Online-tilassa 3D-mallin kanssa. Ohjelmakoodin kokoaminen ei muuttunut juurikaan siitä, miten se tehtiin ilman 3D-mallia, mutta nyt ohjelman simuloinnit osasivat ottaa laskelmissaan ympäristön parametrit tarkemmin huomioon. Jostain syystä ohjelma Grid Loop -malli ei toiminutkaan enää halutulla tavalla, vaan se epäonnistui simuloinneissa kolmannen liikkeen jälkeen ja antoi virheilmoituksen. Tämä vaikutti johtuvan siitä, että jostain syystä ArtiMinds kuvitteli menevänsä pidemmälle kuin mitä oli määritelty. Päättelin näin siitä, että 3D-näkymä näytti aina simuloinnin viimeisimmän liikkeen, mikäli simulointi antoi virheilmoituksen. Kokeilin liikkua toisella komennolla sen pallein reiän kohdalle, josta ohjelma oli antanut virheen. Robotti liikkui oikeaan kohtaan ilman mitään ongelmia erillisellä Move to State -komennolla. Kokeilin tämän jälkeen vielä antaa monta kertaa pallein paikat uudestaan ja katsoa, saako sen toimimaan. Palletointi toimi normaalisti oikein, jos palleista teki pienemmän. Tämäkin viittaisi siihen, että robotti kuvitteli joutuvansa menemään maksimirajojensa yli aiemmassa palleissa ja siksi lopetti liikkeen kesken. Tein lopulta pallein siten, että tein osan käyttäen Grid Loop -palletointitoimintoa, jonka yhdistin toisiin Move to State komentoihin. Kuvassa 12 on nähtävillä ArtiMindsin näkymä valmiin palletointiohjelman visualisaatiosta, jossa on myös eri pisteet ja liikeradat näkyvillä.



KUVA 12. ArtiMinds RPS näkymä 3D-mallin kera ja ohjelmakoodin pisteet nähtävissä



## 5.2 Kokoonpano ja asennustyö BRIO-palikoilla

Yrittäessäni tutkia ja demonstroida voimaohjausta lisää päätimme opettajan kanssa kokeilla niin sanottua palikkatestiä. Palikkatesti on kaikille tuttu lasten lelu, jossa on tarkoitus laittaa oikeanlainen palikka oikeanlaiseen reikään. Tässä tapauksessa kyseessä on vain robotti, joka voimaohjauksella hakee reiät tuntumalla, oikeassa kulmassa ja asennossa tarvittavaan asennukseen. Itse kotelo, johon palikat tulevat, asennetaan työpisteelle noin  $30^\circ$ :n kulmassa. Näin voimaohjausta voidaan demonstroida paremmin, koska robotti joutuu tällöin mittaamaan voimaa useammasta suunnasta yhtäaikaisesti. Tämä tuo lisähaasteita asennuksiin, koska enää ei ole kyse pelkästään sijainnista, vaan myös suunnasta, kulmasta ja voimasta.

Kävin ostamassa kaupasta BRIO:n palikkalaatikon. Laatikossa oli kolmen eri tyyppisiä palikoita ja keräyslaatikko, jossa oli kolme erimuotoista reikää. Asensin ensin laatikon haluttuun kulmaan työpisteelle. Tein tämän käyttäen teippiä ja liimaa, joka ei ole varsinaisiin asennuksiin optimaalinen, mutta tässä tapauksessa kiinnitys oli lähinnä tilapäinen. Sain laatikon pysymään kuitenkin varsin jäämäkästi paikallaan, mikä onkin erittäin tärkeää. Robotti ei voima-anturista riippumatta ollut tunnistaa kosketusta, jos kohde liikkuu helposti johonkin suuntaan. Voima-anturin toimintaperiaate onkin, että robotti liikkuu suuntaan  $x$ , kunnes haluttu voima on saavutettu. Tähän robottiin ei ole liitettyä konenäköä, joten se ei osaa ennustaa, minne kappale liikkuu virheellisen kosketuksen sattuessa. En myöskään tehnyt tästä BRIO-asetelmasta CAD-mallia, vaan käytin pelkästään voimaohjausta. Määritin jokaiselle palikalle siis tarkan hakupaikan, josta ne haetaan. Laatikko ja palikat omilla paikoillaan on nähtävissä kuvassa 13.



KUVA 13. BRIO -kotelo asennettuna kulmaan ja eri palikat kotelon edessä omilla noutopisteissään.

Tässä ohjelmassa käytin robotin liikeratoihin, sekä Path- että Move to State -malleja. Tämä johtui siitä, että kaksi palikoista oli hyvin lähellä robotin nivelten maksimiasentoa. Tämä sai RPS:n tekemään virheilmoituksen, koska se yritti Move to State -toiminnolla nostaa palikkaa nivelten ollessa huonossa nostoasennossa, mikä olisi ollut robotille vaaraksi. Käytin siis käsiohjausta ja Path-malleja näiden kahden palikan kanssa ja nostin näiden avulla ne laatikolle. Move to State -malli oli toki hieman nopeampi, koska sen liikeradat laskee kone ja silloin se käyttää aina suurinta mahdollista maksiminopeutta. Käsiohjauksella liike etenee juuri siihen tahtiin, miten ja millä nopeudella robotia itse ohjaa. Laatikon päällä käytin jokaisen palikan kanssa Spiral Search -voimaohjattuja toimintoja. Olin määrittänyt ainoastaan jokaisen palikan menemään oikeanmuotoisen reiän yläpuolelle. Spiral Search meni itse reiän lähelle ja haki tuntumalla reiän kulmat. Kulman huomattessaan robotti laski palikan syvemmälle reikään ja pudotti sen sinne.

Aluksi minulla oli hieman vaikeuksia saada Spiral Search toimimaan. Tässä huomasin kiinnityksen tärkeyden. Kun laatikko lähti liikkeelle helposti, niin robotti ei saanut siihen hyvää kosketusta ja ohjelma vain lakkasi toimimasta kesken kaiken. Huomasin pian, että voimaohjausta hyödyntävissä Template-malleissa lukee, että malli toimii onnistuneesti, kun/jos robotti tunnistaa kosketuksen

onnistuneesti. Asensin siis laatikon hieman paremmin työpisteen pöytään kiinni, niin että se ei enää liikkunut kosketuksesta, vaan se pysyi paikallaan.

Nyt Spiral Search onnistui kuten sen piti ja robotin tarttuja tunnisti kosketuksen. Ensin palikka osui laatikkoon, mistä robotti tunnisti kosketuksen. Tämän jälkeen se lähti etsimään reiän kulmia pyörivällä liikkeellä kosketusta ylläpitäen. Palikan osuttua reiän kulmaan robotti sijoitti kosketuksen perusteella palikan tarkasti reiän ylle, josta se laski kappaletta sallitun toleranssin mukaan alaspäin oikeassa kulmassa ja lopulta vapautti palikan laatikkoon aukaisemalla tarttujan. Tämän jälkeen robotti haki kaksi muutakin palikkaa ja teki niillä samat toiminnot, kunnes kaikki palikat olivat laatikossa. Toiminnon suorittamista ei haitannut mitenkään se, että palikat ja reiät olivat erimuotoisia.

## 6 TULOKSET

Kummankin ohjelman teko onnistui hyvin, vaikka alkuun olikin pieniä vaikeuksia. Ohjelmat suorittivat tehtävänsä halutulla tavalla ja ne ovat käytännöllisiä eri tuotantoihin. ArtiMinds tuli itselleni tutuksi ja aloin ymmärtämään sen luomia mahdollisuuksia. Voima-anturin sovellutuksista on käytössä läsnäolotunnistus, keskipiste paikannus, käsiohjaus sekä polun tallennus ja näitä sovellettiin palletointiin ja asennukseen. Työn aikana erilaiset kokeilut ohjelmakoodin teossa osoittivat ohjelman vahvuuksia ja heikkouksia. Tärkein tekijä kaikissa mahdollisissa käytettävissä applikaatioissa on, että kappale on joko tukevalla alustalla tai että se on muuten kunnolla kiinnitetty. Voima-anturi tuo robotille paljon paremman tuntuman työstövoimista, kun se on kiinnitettynä. Tästä huolimatta ilman kunnollista kosketusta voimaohjaus ei ole mahdollista. Kun robotti saa halutussa sovellutuksessa tarpeeksi hyvän kosketuksen, voimaohjaus onnistuu.

ArtiMinds osoittautui todella käteväksi ohjelmointiohjelmaksi. Vaikka ohjelma oli minulle aluksi täysin tuntematon, mielestäni opin sen käytön hyvin nopeasti. Olen itse käyttänyt aiemmin opinnoissani UR3 omaa ohjainta ja tehnyt sillä ohjelmakoodeja. Siihen verrattuna ArtiMinds oli todella paljon nopeampi ja yksinkertaisempi. Ohjelman Wizards-apuohjelmat nopeuttavat ohjelmointia valtavasti. Ohjelma osaa pelkästään robotilla annetun demonstraation ja oikeanlaisen Template-mallin valitsemalla sulkea pois kaikki väärät mahdolliset halutut lopputulokset. Toki Template-malleja on monia samantapaisia, joten juuri oikeanlaisen toiminnon löytämisessä voi mennä pieni hetki. Tämä ohjelmointi on miltei puhdasta imitointia ja se nopeuttaa robotin oppimista.

Robotti onnistui korjaamaan liikkeitään ja tunnistamaan sijaintinsa paremmin voima-anturin kanssa. se paransi ja nopeutti robotin toiminnan optimointia ja sen ohjelmoinnista tuli helpompaa. ArtiMinds tukee mielestäni tämänlaista ohjelmointia vallan hyvin. ArtiMinds ja UR-robotti tai mikä tahansa muu yhteistyörobotti muodostavat yhdessä helposti käytettävän ja joustavan kokonaisuuden, jolla on mahdollista nopeasti ohjelmoida ja automatisoida pieniäkin tuotantoeriä. Yhteistyörobotiikka mahdollistaa markkinoiden avautumisen uusille pientuotannon aloille. Universal Robotsin suurimmat myyntitulot tulevat tällä hetkellä juuri pk-yrityksiltä, koska UR-robottien jousto ja yksinkertaisuus tarjoavat robotiikan etuja halvempaan hintaan kuin suuret teollisuusrobotit. Yhteistyörobotit sopivat moniin toistoa sisältäviin käsityötehtäviin, jotka eivät vaadi jatkuvaa kriittistä päätöksentekoa.

Työ osoittaa, että näillä ohjelmistoilla voidaan rakentaa monipuolisia, joustavia ja tehokkaita yhteistyörobotti ratkaisuja monenlaisten pk-yritysten tarpeisiin.

## 7 POHDINTA

Ihmiset imitoivat toisiaan jatkuvasti. Se on ollut kautta aikojen yksiä parhaimpia evoluution kehittämiä oppimisen työkaluja. Eläinlajista riippumatta imitointi on toiminut opettamisen keinona ja tiedon välittäjänä yksilöltä toiselle. Vanhemmat yksilöt ovat opettaneet nuorempia metsästäämään tai tekemään turvallisen kotipesän tietyllä tavalla. Samaan aikaan kuin ihmiset kivikaudella oppivat käyttämään työkaluja seuraamalla toisiaan, he lisäksi harjoittivat yli-imitaatiota, mikä mahdollisti yhä uusien asioiden keksimisen juuri opitusta tiedosta. Mustekalat ovat maapallomme älykkäimpiä eläinlajeja ei pelkästään sen takia, että ne kykenevät oppimaan asioita erittäin nopeasti, vaan myös siksi, että niillä on kyky manipuloida ympäristöään. Eläin- ja neurotieteilijä Tohtori Graziano Fiorito on muun muassa todistanut mustekalojen oppivan pelkästään havainnoimalla toista yksilöä. (Fiorito & Scotto 1992) Mustekalan A piti ratkaista monimutkainen tehtävä. Mustekala B sijoitettiin sen viereen omaan tankkiinsa katsomaan, miten A ratkaisee tehtävän. Tämän jälkeen mustekalan B piti ratkaista sama tehtävä ja se ratkaisi sen välittömästi, vaikka tehtävän järjestelyä vaihdettiin useamman kerran. (A demonstration of an octopus learning through observation 2011)

Tämä mielestäni osoittaa imitaation ja havainnoinnin olevan erittäin hyvä oppimisen keino ja sanoisin niiden olevan tehokkaimpia oppimisen keinoja. Olen itse myös kokenut varsinkin urheilun puolelta, miten havainnoimalla ja imitoimalla voi oppia nopeasti uusia asioita. Olisi paljon helpompaa, jos monimutkaisten koodien kirjoittamisen sijaan voidaan vain näyttää, miten jokin asia tehdään. Tähän ArtiMinds mielestäni juuri pyrkii. Tämä poistaa kalliiden eksperttien tarpeen, koska miltei jokainen kykenee ohjelmoimaan joitain yksinkertaisia sovellutuksia ohjelman avulla jopa pienimuotoiseen tuotantoon. Tämä luo paremmin mahdollisuuksia pk-toimijoille ottaa robotiikka osaksi tuotantoaan. Pelkästään konenäön ja voima-anturien yhdistäminen tuo jo hyvin paljon lisämahdollisuuksia, koska tällöin robotilla on tavallaan näkö- ja tuntoaisti.

Tekoälyn kehityksen alkuvaiheessa tutkijat olivat varmoja, että meillä olisi varsin pian tietoinen älykäs tekoäly. Hyvin nopeasti huomattiin, että ongelmana ei ollut niinkään siinä, mitä tekoälyn pitäisi huomioida, vaan mitä sen ei pitäisi huomioida. Maailma esittäytyy loputtomana määränä mahdollisuuksia, mutta kaikki ovat epäolennaisia, koska itse ei voi osallistua sen toimintaan mitenkään. Kehomme sisältää monia evoluution aikana kehittyneitä aisteja, joiden avulla saamme ympäristöstä dataa, jonka perusteella voimme toimia ympäristössämme turvallisesti. Keho asettaa toimintarajoitteita ja mahdollisuuden manipuloida ulkoista ympäristöä. Ihmisen aivot pyrkivät jatkuvasti määrittelemään kaikki ne asiat, mitä tällä hetkellä ei pidä huomioida, koska ei ole järkevää

tuhlata arvokkaita energioresursseja mihinkään turhaan. Sen mikä on turhaa, määrittelee se, mitä on sillä hetkellä tekemässä, sekä ihmisen arvot yleensäkin. Näemme maailman meidän arvojen ja motivaatioiden kautta. Tietyissä tilanteissa kykenemme hävittämään muun ulkomaailman ja keskittymään vain yhteen asiaan. Aistimme häivyttävät ulos kaiken mikä on epäolennaista tietyn tehtävän suorittamiseen, jos se ei ole välttämätön uhka turvallisuudelle. Aistimme siis häivyttää kaiken epäolennaisen ja vahvistaa hyvin pienen alueen aistimuksia, koska se on energiatehokasta. (The Monkey Business Illusion 2010) Tämä vahvistaa ajatusta siitä, että pyrimme aina poistamaan kaiken turhan, jotta voisimme tehdä tietyn asian mahdollisimman hyvin. Tämän vuoksi imitaatio on todella nopea tapa opettaa robottia, koska se tallentaa liikeradat ja tehtävät tietokantaansa yhdellä kertaa. Antamalla robotille Template-mallin ja näyttämällä sen liikeradat, se osaa sulkea pois muut halutut lopputulokset. Robotti nyt osaa siis häivyttää ulos kaikki muut mahdolliset faktat ympäristöstä ja se keskittyy vain tämän yhden tehtävän suorittamiseen. Robotin liike katkeaa ainoastaan, jos jotain odottamatonta turvallisuutta vaarantavaa tapahtuu. Robotti on myös kykenevä oppimaan uudestaan yhtä nopeasti, jos vahingossa opettaa sitä väärin. Näiden ominaisuuksien avulla niiden ohjelmoiminen ja opettaminen voivat tällä tavalla olla äärimmäisen nopeaa. Uskon tämän ohjelmointitavan olevan yhä enemmän käytössä tulevaisuudessa, koska kautta aikojen imitointi on tuntunut olevan erittäin toimiva vaihtoehto. Ihmisten ja eläinten oppiminen on tapahtunut imitoinnin, havainnoinnin, vanhempien vaikutuksen ja opettamisen kautta. (Gariépy, Watson, Du, Xie, Erb, Amasino, Platt 2014) Tieteellisen ajattelumallin mukaan on myös niin, että jos jokin asia toistuu tietyllä tavalla useissa toisistaan riippumattomissa paikoissa, sen todellisuuden todennäköisyys kasvaa.

On myös huomattu, että mitä enemmän ymmärretään ihmisen toimintaa, sitä kehittyneempiä tekoälyjä voimme rakentaa. Biologisten neuroverkostojen ja kybernetiikan välillä on huomattu monia yhtäläisyyksiä. Useasti robottien toimilla yritetään imitoida ihmisen tekemiä tehtäviä ja liikkeitä. UR3 on myös tehty nimenomaan muistuttamaan ihmiskättä ja sen toimintaa. Tästä syystä onkin hyvä tutkia sitä, miksi ihmiset toimivat, kuten he toimivat. Yhtäläisyyksiä on nähtävissä muihin lajeihin jatkuvasti ja monia psykologian käyttäytymistutkimuksia tehdään rotilla, koska niiden käyttäytyminen muistuttaa ihmisen käyttäytymistä riittävästi.

World Economic Forumin ”The Future of Jobs report 2018” -raportin mukaan automaation ja robotiikan uskotaan vähentävän 75 miljoonaa työpaikkaa seuraavan neljän vuoden aikana, mutta samalla luovan 133 miljoonaa uutta työpaikkaa. (World Economic Forum 2018, 8) Robotiikalle ja työtehtävien automatisoinnille uskotaan siis olevan paljon tarvetta ja mielestäni juurikin imitointi tulee olemaan yksi merkittävimmistä tekijöistä tuotannon tehostamisessa. Antureiden yhdistely

monilla tavoin uusiin robotteihin on tuonut uskomatonta kehitystä robotteihin. Boston Dynamicsin uudet robotit kykenevät avaamaan ovia, liikkumaan monenlaisilla alustoilla ja hyppimään esteiden yli. Heidän uudella Atlas humanoidirobotillaan on 28 niveltä, LiDAR- ja stereonäkö ja se kykenee tekemään voltteja ja parkouria. (Boston Dynamics 2019) Lisäksi Open Bionics -yritys on kehittänyt proteesikäsiä, joissa pinsettiote on mahdollinen. (Open Bionics 2019) (Engineers Created A New Bionic Arm That Can Grow With You 2018) Samankaltaisen käden asentaminen robotille toisi sille valtavasti lisämahdollisuuksia, koska sen näppäryys parantuisi huomattavasti. Tämä mielestäni osoittaa vahvaa robotiikan kehitystä ja osoittaa anturien tuovan lisämahdollisuuksia robotiikan sovellusaloihin, kun eri antureita ja laitteita aletaan luovasti yhdistellä.

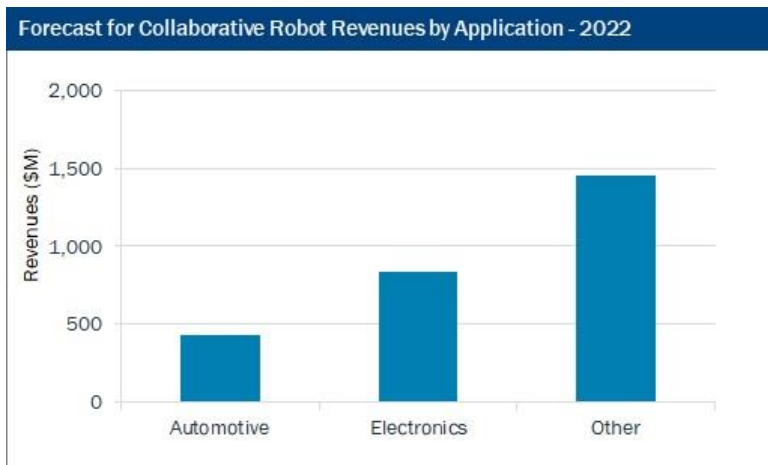
Roboteilla tulee olemaan kilpailuetu ihmisiin nähden. Ne ovat investoinnin ja integroinnin jälkeen valmiita tekemään töitä vaikka kellon ympäri. Ne ovat monesti myös paljon nopeampia, tarkempia, toimintavarmempia ja turvallisempia kuin ihmiset. Robotit voidaan rakentaa samalle alustalle ja ohjelmoida toimimaan aina samalla tavalla. Ihmiset ovat geeneiltään ja persoonaniltaan aivan erilaisia. Reagoimme eri asioihin ja ärsykkeisiin täysin eri tavoin. Pelkästään reaktiomme eri ruoka-aineisiin ovat erilaista. Tämä tekee yksilöiden välisen opettamisen paljon vaikeammaksi kuin mitä roboteilla. Uusien tuotanto- ja tietojärjestelmien avulla pystymme opettamaan robotteja todella nopeasti ja ne oppivat jatkuvasta ajantasaisesta datasta. Jos esimerkiksi tuhat robottia rakennetaan tismalleen samanlaisella alustalla ja ne ovat kaikki yhteydessä samaan verkkoon, yhden robotin oppiessa jotain sen oppivat kaikki loput 999 robottia välittömästi. Tämä nopeuttaa teknologian kehitystä eksponentiaalisesti. Eri anturien ja mittarien yhdistely luovasti erinäisiin robottisoluihin ja tuotantokokonaisuuksiin IoT:n kanssa tulee mahdollistamaan uudenlaiset tehokkaammat tuotannot.

5G-verkon yleistyttyä tiedonsiirto nopeutuu kymmenkertaiseksi, mikä taas mahdollistaa parempien ja kehittyneempien tekoälyjen luomisen. Nopeamman dataliikenteen avulla luodaan taas lisämahdollisuuksia monille edistyneille automatisoiduille sovellutuksille, kuten itseohjautuville autoille ja automatisoidut palvelut. Tämä myös lisää monenlaisen robotiikan ja automatisoinnin kysyntää, koska luodaan uusia käyttökohteita. (5G Wikipedia 2019)

Universal Robotsin työntekijä Stuart Shephard sanoo blogikirjoituksessaan meidän siirtyneen yhteistyörobottien aikaan. Hänen mukaansa yhteistyörobottien ainut ongelma on niiden markkinointi, koska niiden tuomat uudet pientuotannon sovellusmahdollisuudet voisivat kasvattaa asiakassegmenttiä valtavasti. Moni ei ole tietoinen yhteistyörobottien mahdollistamista uusista tuotantomahdollisuuksista ja eivät tästä syystä investoi niihin. (Shephard 2019) Yhteistyörobottien sovellusten aloittaisen myyntiennusteen mukaan elektroniikka- ja autoteollisuuden odotetaan olevan yksittäisiä isoja sovelluskohteita. Kuitenkin muiden tuotannon alojen odotetaan olevan yleisempi

yhteistyörobottien sovelluskohde, koska niiden ennustetut myyntitulot ovat suuremmat kuin auto- ja elektroniikkateollisuuden yhteenlasketut ennustetut myyntitulot. (Robotics Tomorrow 2018) Tämäkin osoittaa yhteistyörobotiikalle olevan kysyntää tulevaisuudessa. Tämä on nähtävissä taulukossa 3.

TAULUKKO 3. Ennuste yhteistyörobottien eri applikaatioalojen myyntituloista vuoteen 2022 mennessä (Robotics Tomorrow 2018)



Robotiikka on Suomessa tällä hetkellä jo tärkeässä asemassa monilla teollisuuden aloilla. Robotiikan uskotaan olevan myös tärkeä tekijä kestäväen kehityksen ja kilpailukyvyn parantamisessa. Valtioneuvoston Selvitys- ja Tutkimustoiminnan tekemän raportin ”Robotisaation ja automatisaation vaikutukset Suomen kansantalouteen 2030” mukaan Suomessa myös tullaan näkemään työpaikkojen häviämistä, mutta ongelmana tulee olemaan vielä monilla aloilla pitkään se, miten robotit saadaan toimimaan tarpeeksi luotettavasti. Tuottavuuden uskotaan kasvavan 2,6 % vuodessa, mutta sen ei uskota vaarantavan nykyisiä työpaikkoja. Yritysten kilpailukyky kasvaa ja samalla henkilömäärällä kyetään jatkuvasti tuottamaan enemmän tuotantoa. (Ventä, Honkatukia, Häkkinen, Kettunen, Niemelä, Airaksinen, Vainio 2018, 87) Raportin mukaansa Suomeen odotetaan yhteistyörobotiikan kysynnän kasvua.

Digitalisaatio vaikuttaa ihmistyön tarpeeseen sekä ruumiillisen että aivotyön osalta. Useita asiantuntijatehtäviä voidaan automatisoida ainakin osittain ja jättää ihmisille se osuus, jota ei kyetä automatisoimaan. Ruumiillisen työn korvaamisesta automaatiolla ja robotiikalla on tehty jo pitkään ja kehitys jatkuu samaa rataa. Ajankohtaisena kehityskulkuna näyttävät olevan robotit, jotka toimivat niin turvallisesti, että ne voivat työskennellä yhdessä ihmisten kanssa sen sijaan, että ne jouduttaisiin sijoittamaan turvahäkkeihin. (Ventä ym. 2018, 19-20)

Suosittelena Centrian panostavaan tähän teknologiaan, koska sillä on selkeästi tulevaisuudessa kysyntää. Uskon tämän teknologian myös olevan käytössä yhä enemmän tulevaisuudessa ja juuri



esimerkiksi tässä työssä käytetyillä laitteilla voisi helposti tuoda lisätehoa omaan pk-tuotantoon. UR-robottien sijoitetun pääoman tuotto on todella hyvä. Tuote maksaa siihen tehtyä investointia takaisin hyvin nopeasti, koska sen integrointi on niin helppoa. Mielenkiintoista olisi lisätä tähän kokonaisuuteen konenäkö, koska se loisi uusia toiminnan lisämahdollisuuksia.

## LÄHTEET

A demonstration of an octopus learning through observation 2011 Viitattu: 17.3.2019 (<https://www.youtube.com/watch?v=GQwJXvITWDw>)

ArtiMinds 2019 Viitattu: 16.3.2019 (<https://www.artiminds.com/>)

ArtiMinds Robotics, ArtiMinds RPS ([https://www.youtube.com/channel/UCCk32K\\_Ouy4Opa5\\_5UXUYzA](https://www.youtube.com/channel/UCCk32K_Ouy4Opa5_5UXUYzA))

ATN, Human-Robot -application 2019 Viitattu: 18.3.2019 (<https://atngmbh.com/en/universal-robots/>)

Boston Dynamics 2019 Viitattu: 17.3.2019 (<https://www.bostondynamics.com/>)

CGTN, Watch an army of robots efficiently sorting hundreds of parcels per hour 2017 Viitattu: 12.3.2019 (<https://www.youtube.com/watch?v=jwu9SX3YPSk>)

Cobot Wikipedia 2019 Viitattu: 17.3.2019 (<https://en.wikipedia.org/wiki/Cobot>)

Cyberneticzoo 2013 Viitattu: 15.3.2019 (<http://cyberneticzoo.com/early-industrial-robots/1954-programmed-article-transfer-patent-george-c-devol-jr-american/>)

Erlbacher E, Force Control Basics, PushCorp Inc. 2000 Viitattu: 19.3.2019 (<http://www.pushcorp.com/pages/technical-papers/Force-Control-Basics.pdf>)

Esineiden Internet Wikipedia 2019 Viitattu: 14.3.2019 ([https://fi.wikipedia.org/wiki/Esineiden\\_internet](https://fi.wikipedia.org/wiki/Esineiden_internet))

Fiorito G & Scotto P, Observational Learning in Octopus vulgaris, 1992 Viitattu: 17.3.2019 [https://www.researchgate.net/profile/Pietro\\_Scotto/publication/6052505\\_Observational\\_Learning\\_in\\_Octopus\\_vulgaris/links/09e41511a52a486581000000.pdf](https://www.researchgate.net/profile/Pietro_Scotto/publication/6052505_Observational_Learning_in_Octopus_vulgaris/links/09e41511a52a486581000000.pdf)

Gariépy J, Watson K, Du E, Xie D, Erb J, Amasino D, Platt M, Social learning in humans and other animals 2014 Viitattu: 17.3.2019 (<https://www.ncbi.nlm.nih.gov/pmc/articles/PMC3982061/>)

Houille M, UR3 Offline Programming, Centria AMK 25.5.2017

Houille M, UR3 Offline Programming with ArtiMinds, Centria AMK 25.5.2017

ISO 8373:2012 Robots and robotic devices – Vocabulary Viitattu: 12.3.2019 (<https://www.iso.org/obp/ui/#iso:std:iso:8373:ed-2:v1:en>)

ISO 10218-1:2011. Robots and robotic devices. Safety requirements for industrial robots. Part 1: Robots. 2011. Viitattu: 12.3.2019 (<https://www.iso.org/standard/51330.html>)

IFR 2017, OptoForce Announces New Model Names and Several New Enhancements to its Two Main Sensor Products 2017 Viitattu: 27.3.2019 (<https://ifr.org/ifr-press-releases/news/optoforce-announces-new-model-names-and-several>)

IFR 2019 Viitattu: 12.3.2019 (<https://www.ifr.org/standardisation>)

Industry 4.0 Wikipedia 2019 Viitattu: 14.3.2019 ([https://en.wikipedia.org/wiki/Industry\\_4.0](https://en.wikipedia.org/wiki/Industry_4.0))

Jäkel R, Schmidt-Rohr S, Stein J, ArtiMinds Robotics GmbH 2015-2016, Expert training in ArtiMinds Robot Programming Suite (RPS), Version 1.0.3 2016, 4 - 80 Viitattu: 16.3.2019

Kompaï Robotics 2019 Viitattu 27.3.2019 (<https://kompairobotics.com/>)

Lempiäinen, J. & Savolainen, J. 2003. Hyvin suunniteltu – puoliksi valmistettu. 1. Painos. Helsinki: Suomen Robotiikkayhdistys Ry, 139 - 142 Viitattu: 19.3.2019

Open Bionics Viitattu: 17.3.2019 (<https://openbionics.com/>)

OptoForce HEX-70-XE-200N pdf (Liite 3 & 4)

Owen-Hill A, Force sensors in robotics research, The essential guide, Robotiq, 2016a Viitattu: 18.3.2019

([https://blog.robotiq.com/hubfs/Force\\_Sensors\\_in\\_Robotics\\_Research.pdf?t=1526460361346](https://blog.robotiq.com/hubfs/Force_Sensors_in_Robotics_Research.pdf?t=1526460361346))

Owen-Hill A, Robotics Research 101: Getting Started with Force Control, Robotiq, 2016b Viitattu: 19.3.2019 (<https://blog.robotiq.com/robotics-research-101-getting-started-with-force-control>)

Pittman K, A History of Collaborative Robots: From Intelligent Lift Assists to Cobots, 2016 Viitattu: 18.3.2019 (<https://www.engineering.com/AdvancedManufacturing/ArticleID/13540/A-History-of-Collaborative-Robots-From-Intelligent-Lift-Assists-to-Cobots.aspx>)

Robotics Tomorrow, Collaborative Robot Market to Grow by 60% in 2018, 2018 Viitattu: 14.3.2019 (<https://www.roboticstomorrow.com/story/2018/10/collaborative-robot-market-to-grow-by-60-in-2018/12661/>)

Salminen P, Robotiikka 2016 Viitattu: 18.3.2019 (<https://zapdoc.site/robotiikka-2016-1-johdanto-ja-historiaa.html>)

Seeker, Engineers Created A New Bionic Arm That Can Grow With You 2018 Viitattu: 17.3.2019 (<https://www.youtube.com/watch?v=luHmXHEpF7w>)

Shephard S, Welcome to the age of cobots 2019 Viitattu: 17.3.2019 ([https://blog.universal-robots.com/welcome-to-the-age-of-the-cobots?utm\\_campaign=Blogpost&utm\\_medium=email&\\_hsenc=p2ANqtz-9X6Jwawj8WyZlopFdE-VNNaPz4czBrrSMKxCIIToFMMy65j1izkQpUubJ31-Tkv7tQOwcoWweanONFmAcvUab5jTI4FtiiNO2dcfh5gGHqkOcT6svQ&\\_hsmi=69639991&utm\\_content=69639991&utm\\_source=hs\\_email&hsCtaTracking=8c01fa74-9680-4616-a131-0d17fc8a335f%7C5046270b-d204-44d9-9172-76d2f6ab6084](https://blog.universal-robots.com/welcome-to-the-age-of-the-cobots?utm_campaign=Blogpost&utm_medium=email&_hsenc=p2ANqtz-9X6Jwawj8WyZlopFdE-VNNaPz4czBrrSMKxCIIToFMMy65j1izkQpUubJ31-Tkv7tQOwcoWweanONFmAcvUab5jTI4FtiiNO2dcfh5gGHqkOcT6svQ&_hsmi=69639991&utm_content=69639991&utm_source=hs_email&hsCtaTracking=8c01fa74-9680-4616-a131-0d17fc8a335f%7C5046270b-d204-44d9-9172-76d2f6ab6084))

Simons D, The Monkey Business Illusion 2010 Viitattu: 27.3.2019

([https://www.youtube.com/watch?v=IGQmdoK\\_ZfY](https://www.youtube.com/watch?v=IGQmdoK_ZfY))

SolidWorks Wikipedia 2019 Viitattu: 17.3.2019 (<https://fi.wikipedia.org/wiki/SolidWorks>)

Spectrum, Robosoft unveils Kompai robot to assist elderly disabled, 2010 Viitattu: 17.3.2019 (<https://spectrum.ieee.org/automaton/robotics/medical-robots/robosoft-kompai-robot-assist-elderly-disabled>)

Stone W.L, The History of Robotics 2005, Teoksessa Kurfess Thomas R. Robotics and Automation Handbook CRC Press 2005 Viitattu: 18.3.2019 ([https://doc.lagout.org/science/0\\_Computer%20Science/8\\_Electronics%20&%20Robotics/Robotics%20and%20Automation%20Handbook.pdf](https://doc.lagout.org/science/0_Computer%20Science/8_Electronics%20&%20Robotics/Robotics%20and%20Automation%20Handbook.pdf))

TechNavio, Top 21 Industrial Robotics Companies in the World 2018, 2018 Viitattu: 18.3.2019 (<https://blog.technavio.com/blog/top-21-companies-in-the-industrial-robotics-market>)

TM Robotics, Cobots and Industrial Robots, Choose the Right Robot for the Job, 2017, s. 2-3 Viitattu: 17.3.2019 ([https://www.tmrobotics.com/downloads/Collaborative-vs.-Industrial-robot-wp\\_FINAL\\_3-16-17.pdf](https://www.tmrobotics.com/downloads/Collaborative-vs.-Industrial-robot-wp_FINAL_3-16-17.pdf))

Universal Robots, Easy automation with collaborative robots. (<https://www.youtube.com/channel/UCM09iVHDc416V8qLj-qhcWQ>)

Universal Robots, Universal Robots Launches UR3, 2015 Viitattu: 18.3.2019 (<https://www.universal-robots.com/about-universal-robots/news-centre/universal-robots-launches-ur3/>)

Universal Robots, UR Main Product Brochure 2019a Viitattu: 18.3.2019 ([https://www.universal-robots.com/media/1801258/eng\\_199901\\_ur\\_main-product\\_brochure\\_web\\_1.pdf](https://www.universal-robots.com/media/1801258/eng_199901_ur_main-product_brochure_web_1.pdf))

Universal Robots UR3, Universal Robots 2019b Viitattu: 17.3.2019 (<https://www.universal-robots.com/products/ur3-robot/>)

United States Patent, Colgate J, Peshkin M, COBOTS 1999 Viitattu: 18.3.2019 (<https://patents.google.com/patent/US5952796>)

Ventä O, Honkatukia J, Häkkinen K, Kettunen O, Niemelä M, Airaksinen M, Vainio T, Robotisaation ja automatisaation vaikutukset Suomen kansantalouteen 2030, 2018, 19-20; 87 Viitattu: 17.3.2019 ([http://julkaisut.valtioneuvosto.fi/bitstream/handle/10024/161102/47-2018-ROBOFINN\\_raportti\\_.pdf](http://julkaisut.valtioneuvosto.fi/bitstream/handle/10024/161102/47-2018-ROBOFINN_raportti_.pdf))

Wallén J, The history of the industrial robot, 2008, 14-15 Viitattu: 17.3.2019 (<http://liu.diva-portal.org/smash/get/diva2:316930/FULLTEXT01.pdf>)

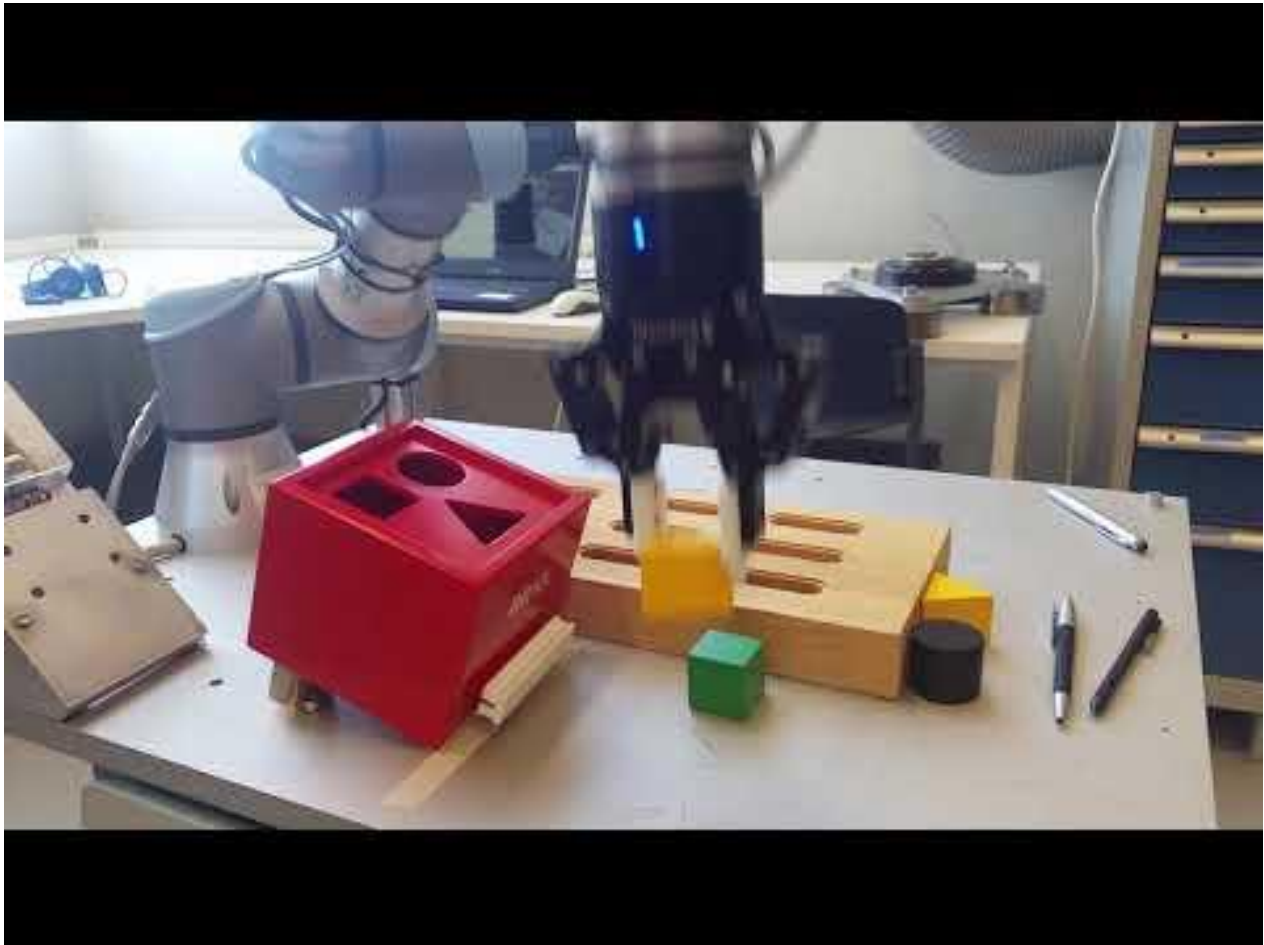
World Economic Forum, The Future of Jobs Report 2018, 2018, 8 Viitattu: 17.3.2019 (<https://www.weforum.org/reports/the-future-of-jobs-report-2018>)

5G Wikipedia 2019 Viitattu: 17.3.2019 (<https://fi.wikipedia.org/wiki/5G>)

## PALLETOINTIOHJELMA



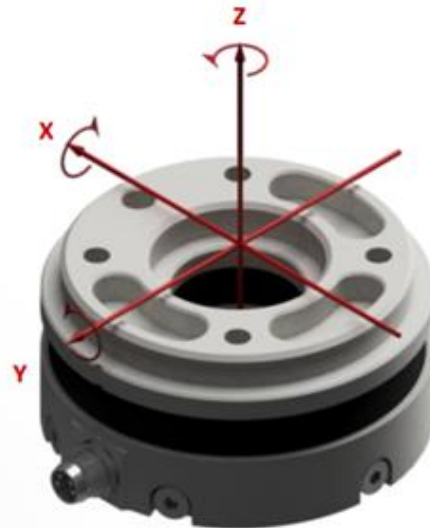
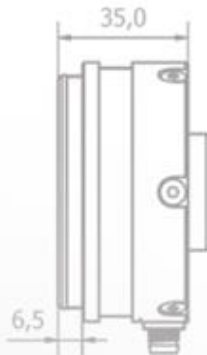
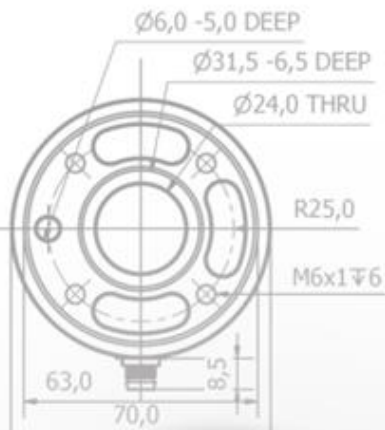
## ASENNUSOHJELMA



OPTOFORCE PDF TEKNISET TIEDOT 1/2



V2.2



**6-Axis  
F/T  
Sensor**

**HEX-70-XE-200N**

**Description:**

OptoForce 6 axis F/T sensors provide 6 degrees of freedom force and torque measurement. Our sensors are designed to fit most of the currently used industrial robot arms. Integration with the various available interfaces is simple. Common applications are **force control** devices, **teach in activities**, and **crash detection**, but the sensors can be used next to **end effectors** in case of **grinding, polishing or deburring** tools. High durability and an unlimited number of custom opportunities resemble all of our sensor types.

We offer these sensors mostly for **system integrator** companies, **robot arm manufacturers** and **contract manufacturing** companies.

**Benefits:**

- Multi axis force measurement
- High resolution
- Highly adaptable product design
- Dust and water proof (IP65)
- High overload range
- Mechanical shock resistant
- Cost efficient solution
- Easy integration

	Nominal Capacity	Deformation (Deflection)
Fxy	± 200 N	± 1.7 mm
Fz	± 200 N	± 0.3 mm
Txy	± 10 Nm	(± 2 °)
Tz	± 6.5 Nm	(± 5 °)

OPTOFORCE PDF TEKNISET TIEDOT 2/2

CAD

DRAWINGS

HEX-70-XE

1: V+  
2: CAN High  
3: CAN Low  
4: V-

Compatible with coupling ISO 9409-1-50-4-M6

SPECIFICATIONS

Sensor Type	6 Axis Force Sensor			
Dimensions	Height x diameter		35 x 70 mm	
Weight	(With built-in adapter plates)		200g	
	Fz	Fxy	Txy	Tz
Nominal Capacity (N.C)	200 N	200 N	10 Nm	6.5 Nm
Single axis overload	200 %	200 %	200 %	200 %
Full scale nonlinearity	2 %	2 %	2 %	2 %
Resolution (typical)	133.33 mN	± 22.22 mN	± 1 mNm	± 0.65 mNm
Single axis deformation at N.C	± 0.3 mm	± 1.7 mm	± 2.5 *	± 5 *
Crosstalk (typical)	< 5 %			
Hysteresis (measured on Fz axis , typical)	< 2 %			
Working temperature range	-40 °C - +80 °C			
Temperature compensation	All axes		-10 °C - +40 °C	
Power requirement	DC input range 7-48V		0.8 W	
Robot mounting screw	4 x M6 x 8 mm		DIN7984-A20-0130	
Sensor mounting screw	6 x M4 x 8 mm		DIN7984-A20-050	

INTERFACE TYPES

USB	CAN	Ethernet - TCP/UDP	EtherCAT
Maximum sampling frequency 1000 Hz			
Supported systems Windows; Linux; ROS; UR			