

Mikko Ylihärsilä

**Robotisoidun hitsauksen etäohjelmointi**

Parametrien hallinnan kehitys ja integrointi etäohjelmointiympäristöön

Opinnäytetyö

Kevät 2010

Tekniikan yksikkö

Kone- ja tuotantotekniikka



## SEINÄJOEN AMMATTIKORKEAKOULU

### OPINNÄYTETYÖN TIIVISTELMÄ

Koulutusyksikkö: Tekniikan yksikkö  
Koulutusohjelma: Kone- ja tuotantotekniikka

Tekijä: Ylihärsilä Mikko

Työn nimi: Robotisoidun hitsauksen etäohjelmointi. Parametrien hallinnan kehitys ja integrointi etäohjelmointiympäristöön.

Ohjaajat: Yli-Suomu Juho & Pakkanen Jarkko

Vuosi: 2010

Sivumäärä: 51

Liitteiden lukumäärä: 3

---

Robottihitsaussolujen määrä on jatkuvasti kasvussa Suomessa. Eräkokojen pienentyessä ja tuotteiden elinkaarten lyhentyessä tuotannon joustavuutta voidaan hakea robotisoinnin ja etäohjelmoinnin avulla.

Tämän työn tarkoituksena oli luoda etäohjelmointimalli Seinäjoen ammattikorkeakoulun konelaboratorion robottihitsaussolusta. Tavoitteena oli saada malli niin tarkaksi että sillä luotuja hitsausohjelmia pystyttäisiin ajamaan muuttamattomina oikeassa solussa. Työssä keskityttiin robotisoituun MIG/MAG-hitsaukseen ja sen etäohjelmointiin. MIG/MAG-hitsauksesta on käsitelty erityisesti hitsausparametreja ja niiden vaikutusta syntyvään hitsiin. Etäohjelmoinnista käsiteltiin sillä saavutettavia hyötyjä tuotteen valmistettavuuteen ja tuotannon ylösajoon.

Robottisolusta tehtiin etäohjelmointimalli Delmia V5 -ohjelmalla. Malli kalibroitiin vastaamaan mahdollisimman tarkasti todellista solua. Ohjelmointikäyttöliittymää kehitettiin niin, että hitsausohjelmia pystyy valmistamaan nopeasti ja helposti. Keskeinen osa työtä oli hitsausparametrien tallentamiseen ja hallintaan kehitetty taulukkopohja, joka integroitiin AMP-hitsausmakroiin. Parametrien hallinta ja hitsausmakrot yhdessä muodostavat hitsauspiirteitä joilla hitsi on täydellisesti toistettavissa. Hitsauspiirteillä käyttäjä pystyy luomaan hitsausohjelmia valitsemalla hitsauskuvan mukaisen hitsityypin ja tarvittavan a-mitan. Työssä kehitetyn etäohjelmoinnin toimintamallin avulla Seinäjoen ammattikorkeakoulun tekniikan yksikkö pystyy tarjoamaan sekä korkeatasoista että ajanmukaista etäohjelmoinnin opetusta ja siihen liittyvää palvelutoimintaa.

Asiasanat: hitsaus, etäohjelmointi, simulointi, kalibrointi, robotiikka

SEINÄJOKI UNIVERSITY OF APPLIED SCIENCES

### **Thesis abstract**

Faculty: School of Technology

Degree programme: Mechanical and Production Engineering

Author: Ylihärsilä Mikko

Title of the thesis: Off-line programming of robotized welding. Development of parameter management and integration to the off-line programming environment.

Supervisors: Yli-Suomu Juho & Pakkanen Jarkko

Year: 2010

Number of pages: 51

Number of appendices: 3

---

The amount of robot welding applications is constantly increasing in Finland. While the batch sizes are reducing and the product life cycles are shortening, the manufacturing flexibility can be achieved with robotizing and off-line programming.

The purpose of this thesis was to create an off-line programming model of the robot welding cell in the Machine Laboratory of School of Technology of Seinäjoki University of Applied Sciences. The goal was to get the model calibrated so accurately that the welding programs made with it could be used without any modifications in the real cell. This thesis focused on the robotized MIG/MAG -welding and its off-line programming. Especially the parameters of MIG/MAG -welding and their effects on the welding seam are covered. Also the advantages of the off-line programming in product manufacturability and production ramp-up are discussed.

The off-line model of the robot cell was made with the Delmia V5 –program. The welding cell model was calibrated to reflect the real cell as accurately as possible. The programming interface was developed to make the welding program preparations fast and easy. The welding parameter management was developed and integrated with AMP –welding macros to form reproducible welding features. With the welding features user can make welding programs by selecting right type of weld and size from the weld drawing. With the off-line programming method developed in this thesis School of Technology of Seinäjoki University of Applied Sciences can offer high quality and modern education and service activities provided for companies.

Keywords: welding, off-line programming, simulation, calibration, robotics

## SISÄLLYS

### TIIVISTELMÄ

### ABSTRACT

### SISÄLLYS

### KÄYTETYT TERMIT JA LYHENTEET

### KUVIO- JA TAULUKKOLUETTELO

<b>1 JOHDANTO .....</b>	<b>9</b>
1.1 Työn tausta .....	9
1.2 Työn tavoitteet.....	9
1.3 Työn rajaus .....	10
<b>2 ROBOTTIHITSAUS .....</b>	<b>11</b>
2.1 Robotisointi .....	11
2.2 Hitsausrobotit ja lisäakselit.....	12
2.3 Railonseuranta ja railonhaku.....	13
<b>3 MIG/MAG-HITSAUS .....</b>	<b>14</b>
3.1 Hitsauslaitteisto .....	14
3.2 Hitsausparametrit .....	16
3.3 Kaarityypit ja aineen siirtymismekanismit.....	16
<b>4 ROBOTTIHITSAUKSEN SIMULOINTI JA ETÄOHJELMOINTI....</b>	<b>19</b>
4.1 Simulointi .....	19
4.2 Kalibrointi .....	19
4.3 Etäohjelmointi .....	20
<b>5 TOTEUTUKSEEN KÄYTETYT TYÖKALUT .....</b>	<b>22</b>
5.1 Delmia V5 .....	22
5.2 Java .....	22
5.3 XML.....	23
5.4 XSLT .....	23
5.5 Xalan-Java .....	23
5.6 VB Script .....	23

5.7 Batch.....	24
5.8 AMP .....	24
<b>6 ETÄOHJELMOINTISOLUN MALLINTAMINEN .....</b>	<b>25</b>
6.1 Mallinnuksen aloitus.....	26
6.2 Hitsauspoltin .....	26
6.3 Pyörityspöytä .....	27
6.4 Ympäristö ja oheislaitteet .....	29
6.5 CAD-mallien tuonti Delmia V5:een.....	30
<b>7 ETÄOHJELMOINTISOLUN KALIBROINTI.....</b>	<b>31</b>
7.1 Kalibroinnin mittavälineet .....	31
7.2 Akselien nollakohtien ja työkalupisteen kalibrointi.....	32
7.3 Työkalupisteen suunnan kalibrointi .....	35
7.4 Pyörityspöydän kalibrointi .....	36
7.5 Työkappaleen kalibrointi .....	36
7.6 Kalibroinnin tarkastus.....	37
<b>8 OHJELMOINTIYMPÄRISTÖN RÄÄTÄLÖINTI .....</b>	<b>38</b>
8.1 Hitsausmakrojen luonti.....	38
8.2 Hitsausparametrien hallinta.....	40
8.3 Parametrien liittäminen hitsauspolkuun.....	41
8.4 Robottikohtaisen kääntäjän muokkaus.....	42
<b>9 OHJELMIEN YLÖS- JA ALASLATAUS .....</b>	<b>44</b>
9.1 Setrobot .....	44
9.2 Makep .....	45
9.3 Printtp.....	45
9.4 Käännöksen automatisointi .....	45
9.5 Ylös- ja alaslatauksen vaiheet.....	46
<b>10CASE SLV .....</b>	<b>47</b>
<b>11TULOKSET JA KEHITYSKOhteet .....</b>	<b>48</b>
<b>12YHTEENVETO JA POHDINTOJA.....</b>	<b>49</b>
<b>LÄHTEET .....</b>	<b>51</b>
<b>LIITTEET</b>	

## KÄYTETYT TERMIT JA LYHENTEET

<b>Tag</b>	Robotin kohdepiste etäohjelmointiohjelmassa.
<b>XML</b>	Extensible Markup Language
<b>XLST</b>	Extensible Stylesheet Language Transformations
<b>AMP</b>	Arcwelding Macro Programming
<b>MIG</b>	Metal Inert Gas
<b>MAG</b>	Metal Active Gas
<b>TIG</b>	Tungsten Inert Gas
<b>WFS</b>	Wire-Feed Speed
<b>STEP</b>	Standardized Exchange of Product
<b>RMS</b>	Root Mean Square
<b>CAD</b>	Computer Aided Design
<b>CAM</b>	Computer-Aided Manufacturing
<b>FMS</b>	Flexible Manufacturing System

## KUVIO- JA TAULUKKOLUETTELO

KUVIO 1. Robottihitsauksen käyttöalue. ....	12
KUVIO 2. Virtalähde pitää kaarenpituuden vakiona. ....	15
KUVIO 3. Suojakaasun vaikutus tunkeumaan. ....	16
KUVIO 4. Siirtymismekanismit. ....	17
KUVIO 5. Delmia V5. ....	21
KUVIO 6. IGRIP. ....	21
KUVIO 7. Mallinnettava solu. ....	25
KUVIO 8. Hitsauspoltin ja malli. ....	27
KUVIO 9. Pyörityspöydän isometrinen kuvanto. ....	28
KUVIO 11. Pyörityspöydän malli. ....	29
KUVIO 12. Piikkimatto kiinnitettynä kalibrointia varten. ....	32
KUVIO 13. Piikkien kohdistus. ....	33
KUVIO 14. Esimerkki kalibrointidatasta. ....	34
KUVIO 15. Kalibroitu robottisolun. ....	37
KUVIO 16. Hitsausmakron suorittaminen. ....	39

KUVIO 17. Hitsausmakron valinta.....	39
KUVIO 18. FLAT_FILLET-makron parametrivälilehti. ....	40
KUVIO 19. Parametrien koontivälilehti.....	41
KUVIO 20. Hitsausparametrien lisäys robottiohjelmaan liikekäskyjen alle.....	42
KUVIO 21. Parametrin lukeminen muuttujaan. ....	42
KUVIO 22. Hitsauksen ja aloituksen käsittely. ....	43
KUVIO 23. Vaaputuksen aloituksen määrittely. ....	43
KUVIO 24. Ohjelma Fanuc-robotille käännettynä. ....	43
KUVIO 25. Robottiohjelmien ylös- ja alaslatauksen vaiheet. ....	46
TAULUKKO 1. Akselien rajat ja nopeudet. (Fanuc 2007)..	29



# 1 JOHDANTO

## 1.1 Työn tausta

Opinnäytetyön aihe on peräisin Seinäjoen ammattikorkeakoulun tekniikan yksiköltä. Työn tarkoituksena on kehittää ammattikorkeakoulun konelaboratorion robotisolun hitsaus etäohjelmoitavaksi samalla tutkia Delmia V5 -ohjelmiston soveltuvuutta robottihitsauksen etäohjelmointiin.

Robottihitsaussolujen määrä on jatkuvasti kasvussa Suomessa ja muualla maailmassa. Eräkokojen pienentyessä ja tuotteiden elinkaarten lyhentyessä tuotannon joustavuutta voidaan hakea etäohjelmoinnin avulla. Tekniikan yksikkö on jo useamman vuoden tarjonnut simulointipalveluja, mutta varsinaista etäohjelmointia ei ole vielä tehty.

Tekniikan yksikön opetuksessakin käytetty Delmian IGRIP -ohjelma on väistymässä ja tämä opinnäytetyö voisi tarjota suoraan hyödynnettävää materiaalia uuden ohjelman opetukseen.

Tällä hetkellä konelaboratoriossa tehdään menetelmätestausta robottihitsauksesta yritysasiakkaille. Opinnäytetyön avulla testausta voidaan nopeuttaa. Täsmällisten hitsausratojen muodossa testauksen kannalta oleelliset hitsausparametrit saadaan pysymään vakaampina.

## 1.2 Työn tavoitteet

Työn tavoitteena on saattaa Seinäjoen ammattikorkeakoulun tekniikan yksikön konelaboratorion hitsausrobotti etäohjelmoitavaksi. Tämä tarkoittaa käytännössä sitä, että simulointimalli saadaan kalibroitua niin tarkaksi, että ohjelmia voidaan

ajaa muokkaamattomina robotilla. Lisäksi ohjelmointiympäristöä on tarkoitus kehittää niin, että hitsausohjelmien teko onnistuu helposti ja nopeasti. Tavoitteena on myös kehittää hitsausparametrien hallintaan työkalu ja integroida se hitsausratojen luontiin, jolloin hitsi voidaan luoda yhdellä hitsauspiirteellä, eikä hitsausparametreja tarvitse syöttää käsin.

### **1.3 Työn rajaus**

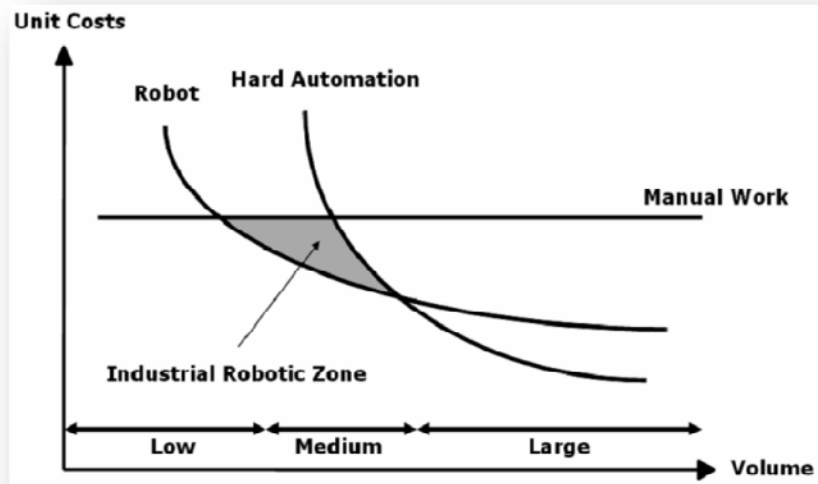
Hitsausrobotti on osa konelaboratoriossa sijaitsevaa FMS-solua, johon on liitetty hyllystöhissi, koneistuskeskus, särmäyspuristin, kaksi robottia ja pyörityspöytä. Työ rajataan niin, että mallinnettavaan etäohjelmointisoluun liitetään varsinainen hitsausrobotti ja pyörityspöytä. Muuta osaa solusta mallinnetaan staattiseksi malliksi törmäystarkasteluja varten. Parametrien hallinnalle ja hitsauspiirteiden valmistukselle luodaan pohja, mutta niiden valmistus rajataan pois runsauden vuoksi.

## 2 ROBOTTIHITSAUS

Robottihitsaus kattaa kaikki hitsaamalla tapahtuvat liittämismenetelmät. Eri menetelmiä ovat muun muassa pistehitsaus, MIG/MAG-kaarihitsaus, TIG-kaarihitsaus, kitkahitsaus, laserhitsaus ja puikkohitsaus. Näistä yleisin robotisoitu menetelmä on MIG/MAG-kaarihitsaus (Bölmsjo, Loureiro & Pires 2006, 73). Etäohjelmoitavassa robottisolussa on MIG/MAG-laitteisto, joten tässä työssä keskitytään MIG/MAG-kaarihitsaukseen.

### 2.1 Robotisointi

Nykyisillä markkinoilla tuotteiden elinkaaret ovat lyhentyneet ja laatua vaaditaan entistä edullisemmin. Yritykset eivät ota riskejä pitämällä suuria varastoja, jolloin valmistettavat erät ovat kohtalaisen pieniä. Tällä alueella roboteilla saavutetaan pienin kustannus valmistettua tuotetta kohden, kuten kuviosta 1 selviää. Robottihitsauksen kannattavuuteen vaikuttaa lisäksi myös hitsattavien liitosten osuus tuotteessa. Jos hitsiliitosten osuus on suuri, tulisi jonkinasteista automatisointia käyttää. Robotisointi lisää kuitenkin uusia haasteita, jolloin tarvitaan taitavampaa henkilökuntaa. Tämä asettaa vaatimuksia robotin ohjelmointikäyttöliittymälle, jotta järjestelmän joustavuus todella saataisiin hyödynnettyä. (Bölmsjo, Loureiro & Pires 2006, 1–2, 17, 22.)



KUVIO 1. Robottihitsauksen käyttöalue. (Bölmsjo, Loureiro & Pires 2006, 2.)

Robotisoinnin kautta yritys voi myös antaa itsestään modernin vaikutelman, joka houkuttelee sekä työntekijöitä että yhteistyökumppaneita. Robotisoinnin on todettu lisäävän työiihtyvyyttä, koska roboteilla korvataan usein raskaat ja vaaralliset työt. Lisäksi robotisoidut työpisteet ovat yleensä siistejä. (Aaltonen & Torvinen 1997, 151,153.)

## 2.2 Hitsausrobotit ja lisäakselit

Suomessa kaarihitsausrobotit otettiin käyttöön vuonna 1979 (Aaltonen & Torvinen 1997, 161). Tällä hetkellä Suomessa on noin tuhat kaarihitsausrobotia ja vuonna 2008 uusia asennettiin 77 kappaletta (Teollisuusrobottilasto 2007) (Teollisuusrobottilasto 2008).

Kaarihitsausrobotit ovat yleensä kuusiakselisia nivelvarsirobotteja, joiden liikeradat ovat monipuolisia ja polttimen asento vapaasti määrättävissä. Hitsauspoltin on melko kevyt, joten kaarihitsausroboteissa panostetaan käsittelykyvyn sijasta tarkkuuteen. Robotin ulottuvuutta voidaan parantaa lineaariradalla ja hitsattavan kappaleen asentoa hallitaan usein yksi- tai kaksiakselisella pyörityspöydällä. Myös useampiakselisia kappaleenkäsittelijöitä kuten toista robotia voidaan käyttää. (Bölmsjo, Loureiro & Pires 2006, 23.)

### 2.3 Railon seuranta ja railonhaku

Railon seuranta korjaa ohjelmoitua hitsausrataa hitsauksen aikana, jos kappale eroaa ohjelmointiin käytetystä kappaleesta. Yleinen railon seuranta menetelmä on käyttää vaaputusta hitsauksen aikana ja mitata kaarijännitteen ja virran vaihtelua. Vaaputuksessa langan etäisyys kappaleesta vaihtelee, jolloin railon keskikohta voidaan tunnistaa. Jotta seuranta toimisi, hitsauksen on lähdettävä railosta. Tällöin voidaan käyttää railonhakua. Railonhaussa hitsauslangalla kosketetaan hitsattavia seinämiä ja radan aloitusta korjataan kosketuspisteistä lasketulla arvolla. Monipalkohitsauksessa ensimmäisen palkon rata tallennetaan robotin muistiin ja se otetaan huomioon hitsattaessa seuraavia palkoja. Vaaputustarpeesta johtuen menetelmä ei sovi ohuille kappaleille, jos halutaan kapea sauma. (Bölmsjo, Loureiro & Pires 2006, 108–109.)

Toinen tapa on käyttää optista anturia, jossa hitsausrailoa skannataan lasersäteellä. Menetelmän etuna on, että hitsauksen aikana voidaan huomioida muutokset railon tilavuudessa, muutokset hitsausraon koossa, silloitushitsaukset sekä virheet kappaleiden asennoissa. Lisäksi hidasta railonhakua ei tarvitse suorittaa. Anturi pitää kuitenkin kiinnittää hitsauspolttimeen, jolloin se vaikeuttaa ratojen luontia ja ahtaisiin paikkoihin pääsyä. Optiset anturit ovat tarkkoja, mutta vielä kalliita. (Bölmsjo, Loureiro & Pires 2006, 79–84.)

### 3 MIG/MAG–HITSAUS

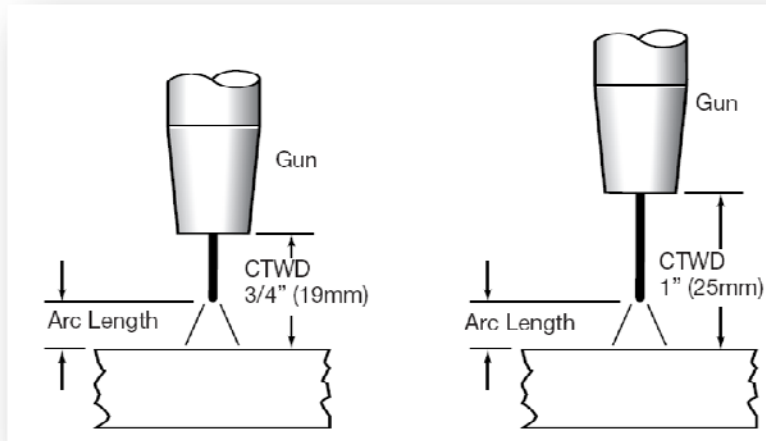
Hitsauksella tarkoitetaan yleisesti metallien liittämistä toisiinsa, mutta myös muita materiaaleja voidaan hitsata. Tässä luvussa käsitellään MIG/MAG-hitsausta erityisesti robotisoinnin kannalta.

MIG/MAG-hitsauksessa syötetään elektronina toimivaa lisäainelankaa hitsauspolttimen läpi työkappaleeseen, jolloin syntyy valokaari. Valokaari sulattaa lisäaineen ja perusaineen hitsisulaksi, jota suojataan suojakaasulla. Kun käytetään reagoimatonta kaasua, prosessia kutsutaan MIG-hitsaukseksi ja kun käytetään reagoivaa kaasua, prosessia kutsutaan MAG-hitsaukseksi. MIG-hitsaus kehitettiin 1940-luvun lopulla raudattomien metallien, kuten alumiinin, hitsaukseen. Menetelmä mahdollisti suuren hitsintuoton, mutta suojakaasujen hinta esti vielä sen käytön terästen hitsaukseen. Vuonna 1953 esiteltiin MAG-hitsaus, jossa hiilidioksidi toimii suojakaasuna. Menetelmän suosio kasvoi nopeasti, koska siihen pystyttiin käyttämään MIG-hitsauksen laitteistoa. (Bölmsjo, Loureiro & Pires 2006, 10,36.)

#### 3.1 Hitsauslaitteisto

MIG/MAG-hitsauslaitteisto koostuu virtalähteestä, lisäainelangansyöttöyksiköstä, hitsauspolttimesta ja kaasusäätimestä. Robotisoidussa järjestelmässä voi lisäksi olla hitsauspolttimen huoltoasema. Huoltoasemassa poltin putsataan, lanka katkaistaan oikeaan pituuteen ja suutin sumutetaan roiskeiden tarttumista estävällä aineella. Virtalähde on tyypillisesti invertterityyppinen, joka antaa tasajännitteisen ulostulon. Virtalähde toimii yhdessä langansyöttimen kanssa, jolloin laitteisto osaa kompensoida hitsauspolttimen etäisyydenvaihteluita kappaleen pinnasta säätämällä itseään ja samalla vakavoittaen kaaren pituutta. (Bölmsjo, Loureiro & Pires 2006, 38.)

Nykyisin monipuolisilla virtalähteillä voidaan tuottaa myös pulssihitsaukseen tarvittavaa pulssivirtaa. Alla olevassa kuviossa näkyy, kuinka virtalähde pitää kaaren pituuden vakiona, vaikka polttimen etäisyys ja vapaalangan pituus muuttuisi.



KUVIO 2. Virtalähde pitää kaarenpituuden vakiona. (GMAW Welding Guide 2006.)

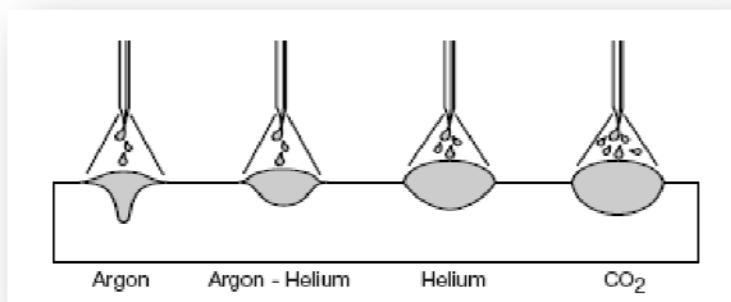
Lisäainelangan syöttöyksikkö syöttää lankaa työntämällä sitä tasaisella nopeudella hitsauskaapeliin läpi hitsauspolttimelle. Työntäminen tapahtuu rullien avulla. Teräslangoille käytetään pyällettyjä rullia ja pehmeille langoille U- tai V-rullia. Väliin harventamiseksi robottisovelluksissa lisäaine syötetään keloilta tai isoista rummuista. Hitsauskaapelin läpi kulkee lisäksi hitsausvirta, jäähdytysvesi ja suoja-kaasu. Kaapeli on normaalisti kytketty ulkoisesti robotin ranteeseen, jolloin se on alttiina törmäyksille ja kulumiselle. Nykyaikaisissa roboteissa kaapeli voidaan johdattaa myös robotin ranteen läpi. (Bölmsjo, Loureiro & Pires 2006, 39-40.)

Hitsauspolttimessa virta siirtyy virtasuuttimesta lisäainelankaan ja kaasusuutin kohdistaa suoja kaasun hitsiin tasaisella virtauksella. Robotisoiduissa sovelluksissa käytetään vesijäähdytteisiä polttimia, joilla voidaan hitsata taukoamatta. Polttimissa on normaalisti jokin hätäpysäytystoiminto törmäyksien varalta, jotta itse robotti ei vahingoittuisi. (Bölmsjo, Loureiro & Pires 2006, 40.)

### 3.2 Hitsausparametrit

MIG/MAG - hitsauksen perusparametrit ovat virta, jännite, hitsausnopeus, vapaa-  
langan pituus ja langan halkaisija. Myös käytetyllä suojakaasulla, lisäainelangan  
koostumuksella ja hitsaus- ja kuljetuskulmalla on merkitystä syntyvään hitsiin.  
Nämä parametrit ovat toisistaan riippuvia. Esimerkiksi jännite on riippuvainen kaa-  
ren pituudesta ja käytetystä suojakaasusta. (Bölmsjo, Loureiro & Pires 2006, 40.)  
Parametreista virtaa ei voida suoraan säätää, vaan se tapahtuu pääasiassa lan-  
gansyöttönopeutta muuttamalla. Jotta hitsi olisi toistettavissa, on tiedettävä myös  
siinä käytetty vaaputus ja hitsausasento. Lisäksi on laitteistokohtaisia parametreja,  
joilla hitsauskaarta voidaan hienosäätää.

Suojakaasu vaikuttaa tunkeuman syvyyteen ja muotoon. Argon ja helium ovat  
passiivisia kaasuja, joita käytetään kevytmetallien hitsaukseen. Heliumilla saavute-  
taan isompi hitsisula, mutta se on argonia kalliimpaa. Hiilidioksidi on reagoivaa  
kaasua, jota käytetään terästen hitsaukseen. Sillä on hyvä tunkeutuvuus, mutta  
roiskeiden määrä on suuri. Kaasuja voidaan käyttää sekä puhtaina, että seoksina.  
Tyypillisiä seoksia on Ar/CO<sub>2</sub> teräksille, Ar/O<sub>2</sub> ruostumattomille, Ar/He ei-rauta  
metalleille. (Bölmsjo, Loureiro & Pires 2006, 40-41.) Alla olevassa kuviossa on  
vertailtu eri suojakaasujen vaikutusta tunkeumaan.



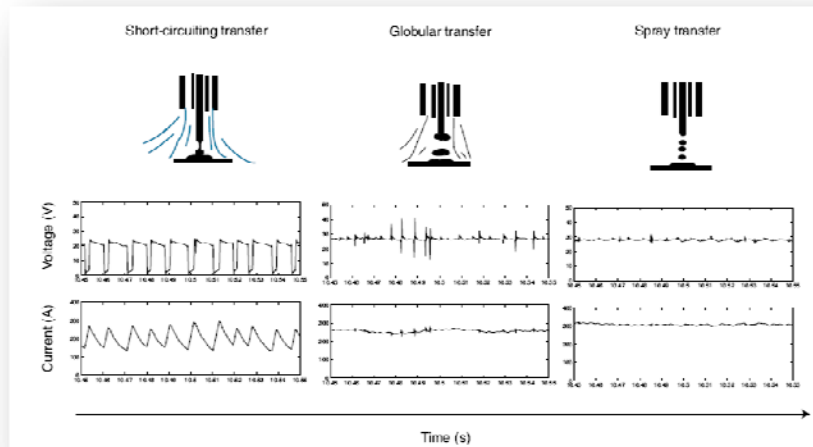
KUVIO 3. Suojakaasun vaikutus tunkeumaan. (GMAW Welding Guide 2006.)

### 3.3 Kaarityypit ja aineen siirtymismekanismit

MIG/MAG -hitsauksessa hitsausparametrit vaikuttavat syntyvän kaaren tyyppiin ja  
aineen siirtymismekanismiin. Aineen perussiirtymismekanismit ovat oikosulku-,



suuripisarainen- ja suihkumainen siirtyminen. Keskimääräisen hitsausvirran kasvassa siirtymistyyppi muuttuu oikosulusta suuripisaraiseen ja suihkumaiseen, kuten kuviossa 4. (Bölmsjo, Loureiro & Pires 2006, 88.)



KUVIO 4. Siirtymismekanismit. (Bölmsjo, Loureiro & Pires 2006, 88.)

Lyhytkaarihitsauksessa aineensiirtyminen tapahtuu oikosulkusiirtymisenä, sekakaarissa suuripisaraisena ja kuuma- ja pulssikaarissa suihkumaisena siirtymisenä. Lyhytkaari syntyy kun käytetään pieniä virtoja ja jännitteitä reagoivan kaasun kanssa. Lyhytkaarihitsauksessa keskimääräinen virta ja täten lämmön tuonti on pienempi kuin kuumakaarihitsauksessa. Tästä syystä lyhytkaarella voidaan hitsata ohuempia materiaaleja, mutta hitsintuottokyky ja tunkeuma on huonompi. Menetelmä sopii hyvin asentohitsaukseen. (Bölmsjo, Loureiro & Pires 2006, 41,92.)

Sekakaari syntyy korkeammilla virran ja jännitteen arvoilla kuin lyhytkaari, jos käytetään passiivista kaasua. Reagoivan kaasun kanssa tarvitaan korkeammat virrat ja jännitteet. Sekakaarissa langan vahvuiset tai sitä suuremmat pisarat siirtyvät pienellä taajuudella saumaan. Painovoimasta johtuen menetelmää voidaan käyttää vain alaspäin hitsattaessa ja roiskeiden määrä on suurta. Yleensä sekakaari ei ole toivottu kaarityyppi, vaan halutaan pysyä joko lyhytkaari tai kuumakaarialueella. (Bölmsjo, Loureiro & Pires 2006, 41,88.) Menetelmä on kuitenkin edullinen, koska voidaan käyttää edullisia suojakaasuja, lisäaineita, laitteistoa sekä käyttää suurta hitsausnopeutta. (GMAW Welding Guide 2006.)

Kun virrat ja jännitteet ovat suuremmat kuin sekakaareissa ja kaasu on argon pitoinen, muuttuu kaari kuumakaareksi. Kuumakaareissa pienet pisarat siirtyvät hitsiin suurella taajuudella. Syntyvien roiskeiden määrä on pieni, hitsisula on suuri ja tunkeutuvuus on hyvä, mutta lämmöntuonti kappaleeseen suurta ja asentohitsaus rajoitettua. Menetelmää käytetään paksuille levyille kun halutaan suuri hitsintuottokyky. (Bölmsjo, Loureiro & Pires 2006, 41.)

Pulssikaareissa aineen siirtyminen tapahtuu myös pieninä pisaroina, mutta keskimääräinen virta on pienempi kuin kuumakaareissa. Lämmöntuonti on täten pienempää ja asentohitsaus helpompaa. Menetelmällä voidaan hitsata myös ohuempia levyjä säilyttäen kuitenkin kuumakaaren hitsintuottokyky. Lämmöntonnin pienentämiseksi pulssihitsauksessa pidetään pientä taustavirtaa ja lisäainepisara irrotetaan virtapulssin avulla. (Bölmsjo, Loureiro & Pires 2006, 41,89-90.)

## 4 ROBOTTIHITSAUKSEN SIMULOINTI JA ETÄOHJELMOINTI

### 4.1 Simulointi

Robottijärjestelmiä suunniteltaessa voidaan järjestelmästä tehdä simulointimalli, jolla testataan järjestelmän toimivuus ja minimoidaan suunnitteluvirheet. Malliin voidaan sijoittaa erilaisia robotteja ja kappaleen hitsausta testata erilaisilla laitteistoilla. Simuloinnilla voidaan tarkastaa myös robotin ulottuvuus jokaiseen hitsiin, jolloin yllätyksiä ei tapahdu ylösajovaiheessa. Samalla selviää tahtiaika hitsattavalle kappaleelle eri laitteistokokoonpanoilla. (Aalto, ym.1999, 96.) Simulointimallia voidaan myöhemmin käyttää etäohjelmointiin, kun malli ensin kalibroidaan. Tämä tarkoittaa käytännössä sitä, että kun robottijärjestelmä on simuloitu suunnitteluvaiheessa, kappaleen hitsausohjelmat ovat jo valmiiksi tehtynä.

### 4.2 Kalibrointi

Kalibroinnilla pyritään saamaan mallinnettu etäohjelmointisolu vastaamaan todellista solua mahdollisimman tarkasti. Kalibroinnilla voidaan korjata geometriset virheet eli robotin, oheislaitteiden ja työkappaleiden oikeat sijainnit. Yleensä kalibroidaan vielä robotin nivelten nollakohtien poikkeamat ja työkalupisteet. Kun mallista halutaan todella tarkka, kalibroidaan lisäksi robotin nivelvarsien pituudet ja nivelten pyörähdysakselien kallistukset. Pienissä robottisoluissa voidaan kalibrointiin käyttää robottia mittalaitteena ja koordinaattimittalaitteella mitattua piikkimattoa, jossa on yleensä kolme tai useampi vastinpiikki. Kalibrointimittaukset tehdään ajamalla robotin kalibrointiapiikki vastinpiikkeihin. Isommissa soluissa, joissa robotti on esimerkiksi kiinni lineaariradalla, on helpompaa ja nopeampaa käyttää ulkoista mittalaitetta. Ulkoinen mittalaite on yleensä tarkempi ja sillä voidaan kerätä suurempi määrä mittauksia myös ympäröivästä solusta kohtuullisessa ajassa. Mittaustieto syötetään etäohjelmointi-ohjelman kalibrointifunktiolle, joka korjaa poikkeamat mallissa. Kalibrointifunktio antaa käyttäjälle palautteen kalibroinnin onnistumisesta

virhemarginaaleina. Kalibroinnin tulos tarkastetaan vielä robotille tehtävällä tarkastus-ohjelmalla ajamalla robottia tarkastuspisteisiin. (Aalto, ym.1999, 104–105.)

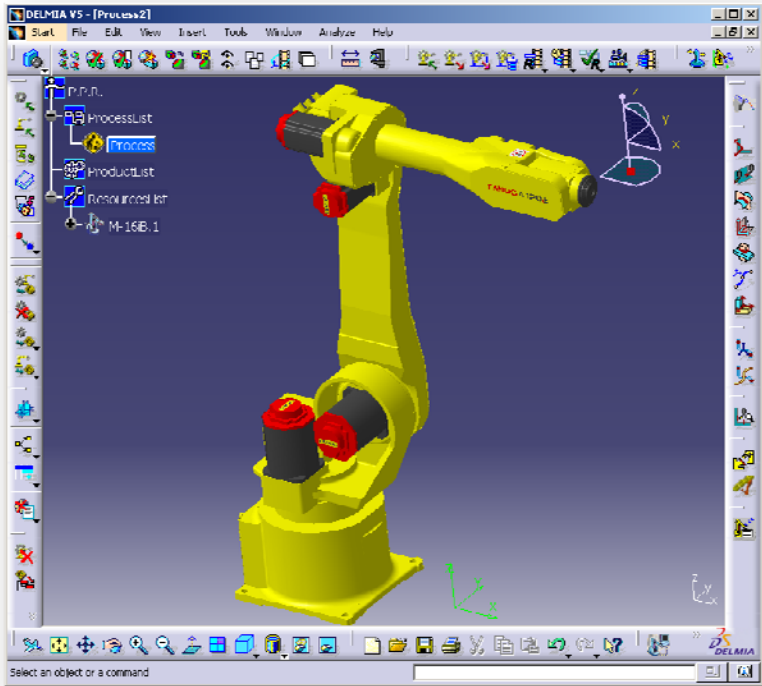
### 4.3 Etäohjelmointi

Etäohjelmointi tarkoittaa robottien ohjelmointia erillään robotista. Ohjelmointi voidaan suorittaa graafisesta käyttöliittymästä käsin ja valmis ohjelma ladataan robotille, jolloin uuden tuotteen ylösajo on nopeaa. Etäohjelmoinnin avulla tuotteen valmistus ja suunnittelu voidaan tehdä yhtä aikaa, jolloin tuotteen valmistettavuus robotilla varmistetaan. (Aalto, ym. 1999, 81–84.)

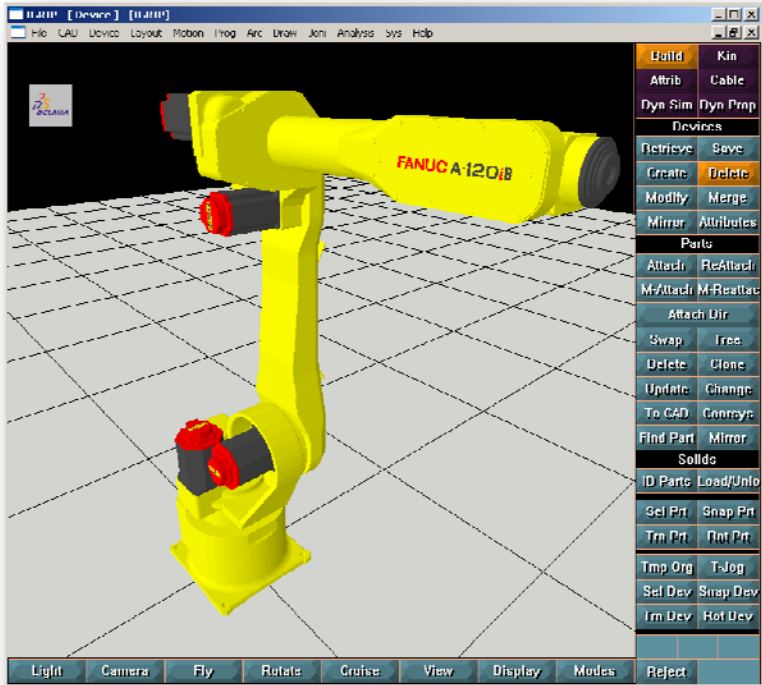
Kappaleen kiinnitin voidaan suunnitella etäohjelmoinnin yhteydessä. Tuotteelle tehdään hitsausohjelma ja kiinnitin suunnitellaan niin, että poltin ei törmää siihen. Tässä vaiheessa havaitaan vaikeasti robotilla hitsattavat kohdat, jolloin tuotteeseen voidaan vielä tehdä muutoksia. (Aalto 2005.)

Etäohjelmoinnilla voidaan saavuttaa taloudellista hyötyä muun muassa tuotantoseisokkien lyhentymisellä ja robotin kaarisuhteen nousemisella. Lisäksi etäohjelmoinnin avulla voidaan tehdä pienempiä sarjoja, jolloin myös asiakasmäärä kasvaa. Tarjouslaskelmista voidaan tehdä tarkemmat, kun tiedetään tuotteen hitsausajat simuloinnin avulla. (Aalto 2005.)

Kuvioissa 5 ja 6 on esitelty kaksi etäohjelmointiin tarkoitettua ohjelmaa, Delmia V5 ja IGRIP. Molemmat ovat yleiskäyttöisiä, eivätkä ole sidottu mihinkään tiettyyn robottimerkkiin. Robottivalmistajilla on myös omia etäohjelmointiohjelmaa, mutta niiden käyttö rajoittuu valmistajan omien robottien ohjelmointiin.



KUVIO 5. Delmia V5.



KUVIO 6. IGRIP.

## 5 TOTEUTUKSEEN KÄYTETYT TYÖKALUT

Opinnäytetyössä hyödynnettiin monia eri työkaluja, johtuen lähinnä niiden läheisestä riippuvuudesta Delmia V5 -ohjelmaan. Työkalujen runsaus toi mukanaan myös paljon haasteita, koska moneen työkaluun liittyi jokin ohjelmointikieli, joka oli kaikissa erilainen. Alla on kerrottu pääpiirteittäin työkaluista ja niiden käyttötarkoituksista.

### 5.1 Delmia V5

Delmia V5 pohjautuu Catia V5 -ohjelmaan, joka on yksi markkinoiden kehittyneimpiä mallinnusohjelmia. Catian käyttäjiin kuuluu esimerkiksi suuria auto- ja lentokoneteollisuuden yrityksiä. Yksi syy tähän on varmasti se, että ohjelmassa on tehokkaat työkalut kaarevien pintojen mallinnukseen. Delmia V5 sisältää käytännössä Catian ominaisuudet, mutta siinä voidaan tuotesuunnittelun lisäksi suunnitella ja simuloida valmistusta. Delmia V5 -ohjelmistoon kuuluu useita eri paketteja, joista tässä työssä tarvittiin Process Detailing & Validation -pakettia.

### 5.2 Java

Java on järjestelmäriippumaton, alun perin sulautettuihin järjestelmiin kehitetty ohjelmointikieli. Järjestelmäriippumattomuudesta johtuen Java tarvitsee virtuaalikoneen tavukoodiksi käännettyjen ohjelmien suorittamiseen. Delmia V5:ssä robotiohjelmien ylöslataamiseen tarkoitetut ohjelmat on kirjoitettu Javalla ja käännetty tavukoodiksi. Näitä ohjelmia kutsutaan parsereiksi ja niiden tehtävänä on lukea annettua robotiohjelmaa ja jäsentää se XML-muotoon.

### **5.3 XML**

XML on formaatti hierarkkisen informaation säilömiseen. Formaatti kehitettiin Web-sivujen tietosisällön tallentamiseen. Formaatti on kuitenkin yleismaailmallinen ja siitä muodostunut eri ohjelmien tiedonjaon väline. Delmia V5 -ohjelmisto käyttää XML:ää moneen eri tiedontallennustapahtumaan, mutta tämän työn kannalta merkityksellisintä on XML-formaattiin tallennettujen robottiohjelmien käsittely.

### **5.4 XSLT**

XSLT on formaatti XML-dokumenttien muuntamiseksi tarkoituksenmukaiseen muotoon. Delmia V5:ssä XML-muotoinen robottiohjelma voidaan kääntää eri robottikielille käyttämällä kyseisen robottikielen XSLT-dokumentilla määriteltyä käännösohjetta.

### **5.5 Xalan-Java**

Xalan-Java on XSLT -prosessori, joka suorittaa varsinaisen muunnoksen XML-formaatista toiseen XSLT-tyylitiedoston ohjeiden mukaisesti. XML- ja XSLT-dokumentit ovat yksistään vain tekstimuotoisia tiedostoja.

### **5.6 VB Script**

VB Script on Visual Basicin kaltainen ohjelmointikieli, joka voidaan helposti integroida eri ohjelmiin. Delmia V5:ssä VB Scriptillä voidaan ohjelmoida makroja laajentamaan ohjelmiston toiminnallisuutta tai helpottamaan usein toistuvien toimenpiteiden suorittamista.

## 5.7 Batch

Batch-ohjelmat ovat tekstimuotoisia tiedostoja, joiden pääte on **bat** tai **com**. Ohjelmilla voidaan suorittaa kaikkia komentokehoteeseen syötettäviä komentoja. Batch-ohjelmia käytettiin useintoistuvien komentokehoteessa tapahtuvien toimenpiteiden automatisointiin.

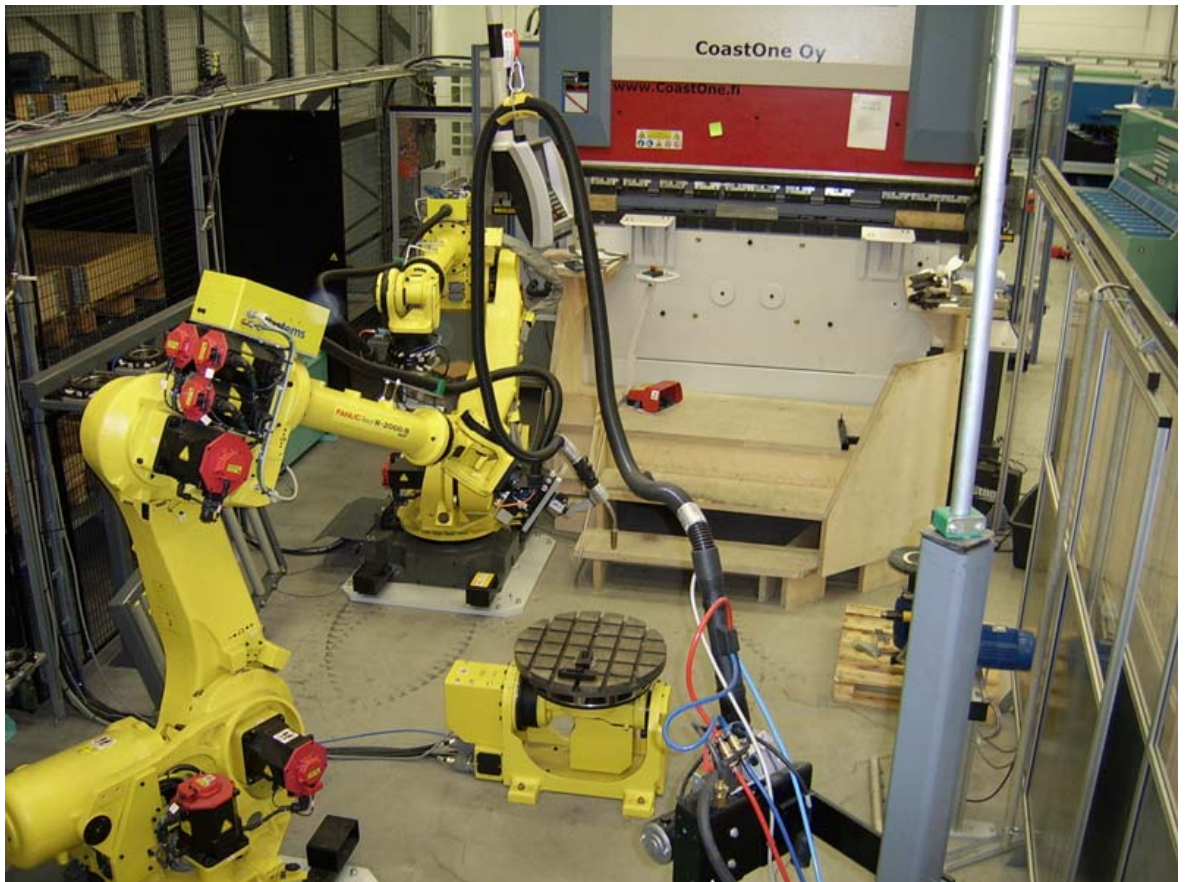
## 5.8 AMP

AMP (Arcwelding Macro Programming) on työkalu monimutkaisten hitsauspolkujen luomiseen. AMP:in vahvuus on siinä, että parametrit voidaan antaa käyttäen standarditerminologiaa. AMP Primitiivit ovat tekstitiedostoja, joissa määritellään muutamilla avainsanoilla parametrinen hitsaussauma. (Online Documentation 2009.)



## 6 ETÄOHJELMOINTISOLUN MALLINTAMINEN

Mallinnettava solu sijaitsee Seinäjoen ammattikorkeakoulun konelaboratoriossa. Mallinnettava kokonaisuus rajattiin niin, että etäohjelmointisoluun liitettiin varsinainen hitsausrobotti ja pyörityspöytä (kuvio 7). Muuta osaa solusta mallinnettiin staattiseksi malliksi törmäystarkasteluihin tarvittava määrä. Mallinnustyökaluna käytettiin Delmia V5 -ohjelmaa.



KUVIO 7. Mallinnettava solu.

## 6.1 Mallinnuksen aloitus

Etäohjelmointisolua mallinnettaessa syntyy paljon tiedostoja, jolloin niiden hallinta voi muodostua hankalaksi. Tämän ehkäisemiseksi solumallille luotiin oma kansio, johon kaikki siihen liittyvä tieto tallennettiin. Lisäksi roboteille, oheislaitteille, mittausdatalle, AMP-tiedostoille, robottiohjelmille ja solussa valmistettaville tuotteille luotiin omat kansionsa. Delmia tarjoaa myös PDM-järjestelmää, joka voi tulla kyseeseen, jos hallittavaa tietoa on paljon.

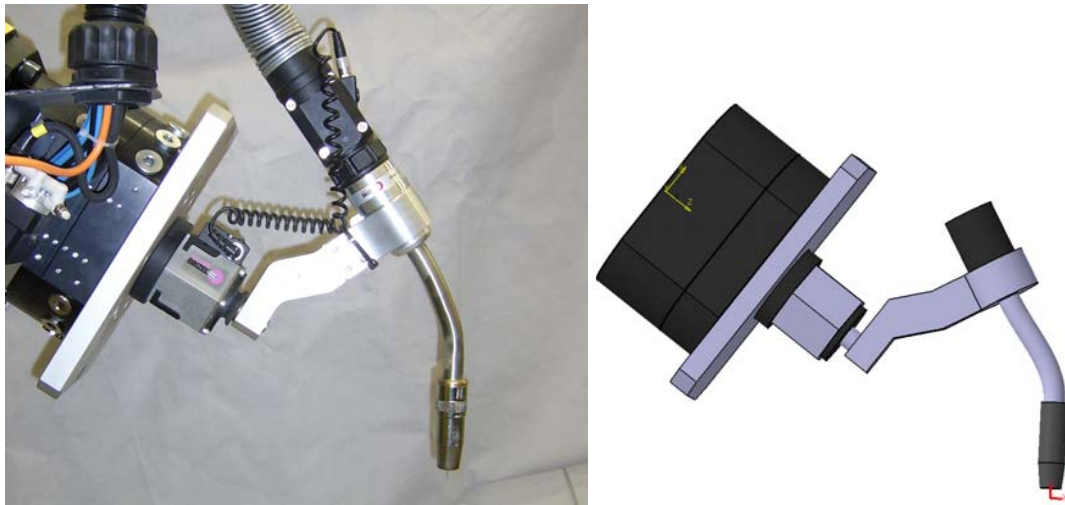
Solumallin rakennus aloitettiin luomalla ensin uusi **Process**-tyyppinen tiedosto, johon koko malli tallentuu. Seuraavaksi malliin lisättiin robottikirjastosta Fanuc 2000iB -robotti. Robotti tallennettiin solumallin kansioon, jotta siihen tehdyt muutokset eivät tallentuisi kirjastossa olevaan alkuperäiseen robottimalliin. Etäohjelmointisoluun kappaleita tuotaessa huomioitiin, ovatko ne tuotteita vai niiden valmistukseen tarvittavia resursseja. Tuotteet ja resurssit näkyvät ohjelman hierarkiassa erillään. Mallinnusta ja kalibrointia tehtiin osittain rinnakkain. Esimerkiksi hitsauspolttimen vapaalangan paikka oli tiedossa jo mallinnusvaiheessa, jolloin mallintaminen oli helpompaa.

## 6.2 Hitsauspoltin

Hitsauspoltin mallinnettiin luomalla uusi tuotemalli. Tuotemalliin lisättiin kaksi uutta osaa, joista toinen kuvaa polttimen runko-osaa ja toinen suutinta. Runko-osaan luotiin koordinaatistoryhmä, jonka alle lisättiin uusi **Base**-tyyppinen koordinaatisto osan origoon. Koordinaatisto kuvaa polttimen kiinnityskohtaa robotin laippaan. Sama toistettiin suutinosalle, mutta siinä koordinaatiston tyyppiksi valittiin **Tool**. Tool-koordinaatisto kuvaa polttimen työkalupistettä, joka on vapaalangan päässä. Lisättyjen koordinaatistojen avulla ohjelma osaa tunnistaa kappaleen työkaluksi, jolloin se voidaan kiinnittää robotin laippaan. Koska malli koostuu kahdesta osasta, runko-osa voidaan pitää paikallaan robotin laipassa ja suutinosaa siirtää oikeaan kohtaan. Tässä vaiheessa suutinosan origo siirrettiin kalibroituun työkalupisteeseen ja poltinmalli vietiin soluun. Poltinmalli kiinnitettiin robotin laippaan **Set Tool-**

komennolla. Osien mallintamista helpotti, kun kaikki koordinaatit olivat oikeissa paikoissa.

Työkalun geometria mitattiin käyttäen avuksi työntö- ja rullamittaa, joilla saatiin riittävä tarkkuus tarvittavaan malliin. Poltinosaan mallinnettiin polttimen suutin ja siitä lähtevä käyrä putki. Muut osat työkalusta mallinnettiin pikakiinnitysosaan. Koska vapaalanka, eli työkalupiste, ei ole täysin kohtisuorassa runkoon nähden, se mallinnettiin kolmiulotteisella spline-käyrällä. Käyrän toinen pää kytkettiin runkoon linkittämällä liitoskohtaan sijoitettu piste suutinosaan. Ennen linkittämistä ohjelman asetuksista täytyy valita **Keep link with selected object** päälle, jotta käyrä seuraisi suutinosan sijainnissa ja asennossa tapahtuvia muutoksia. Jos polttimella jossain vaiheessa törmätään ja siihen syntyy muodonmuutoksia, voidaan muutokset malliin tehdä uusien työkalun kalibrointitulosten avulla siirtämällä ainoastaan suutinosan origo uuteen työkalupisteeseen. Alla on kuva todellisesta polttimesta ja mallinnetusta polttimesta.



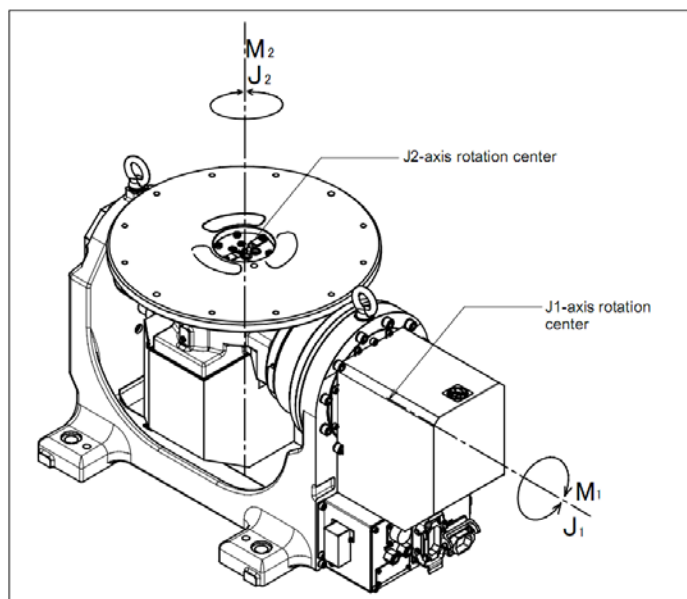
KUVIO 8. Hitsauspoltin ja malli.

### 6.3 Pyörityspöytä

Pyörityspöydän malliin luotiin kolme osaa – runko ja kaksi akseliosaa. Rungon origon sijainniksi valittiin rungon pohja ja sille luotiin koordinaatit ensimmäisen ja toisen akseliosan pyörähdysakseleille. Toinen vaihtoehto olisi ollut sijoittaa rungon

origo suoraan akseleiden leikkauspisteeseen. Tällöin malli olisi kuitenkin ilmaantunut aina puoliksi lattiapinnan alapuolelle solumalliin tuotaessa.

Geometrian mallintamiseksi käytettiin apuna pyörityspöydän huoltokirjassa olleiden profiilikuvien lisäksi työntö- ja rullamittaa. Kuviossa 9 on esitetty pyörityspöytä isometrisessä kuvannossa. Pyörityspöydän päälle kiinnitetty kappaleiden paikointialusta mallinnettiin tarkasti, jotta kappaleen sijoittelua pöytään voidaan suunnitella siinä olevien urien perusteella. Pyörityspöydän akselien liikealueet ja maksiminopeudet luettiin myös huoltokäsikirjasta. Akselien positiivinen pyörimissuunta selvitettiin ajamalla niitä ohjaimesta.

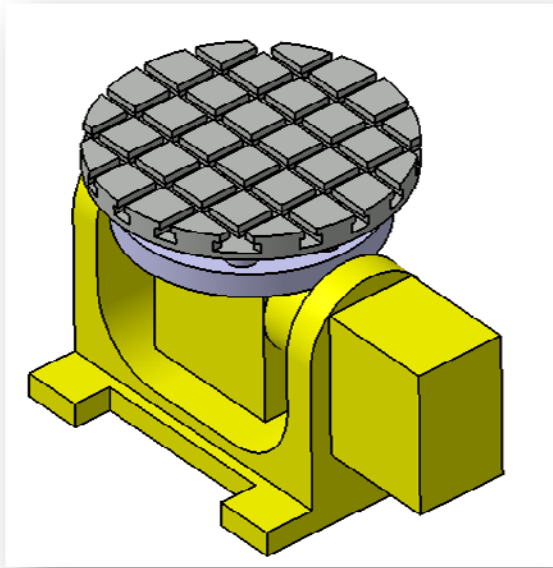


KUVIO 9. Pyörityspöydän isometrinen kuvanto. (2-axes Servo Positioner Maintenance Manual 2007.)

Laitteelle luotiin uusi mekanismi ja sen alle kaksi liikeakselia valitsemalla pyörähdyksakseleiksi äsken luodut koordinaatistot. Molemmille valittiin **Angle driven**-valinta päälle, jolloin akseleita voitiin liikutella. Akseleita liikuteltiin ja tarkastettiin, että niiden pyörimissuunnat olivat oikein. Koordinaatistoja käännettiin tarvittaessa, jos suunta oli väärin. Akselien liikealueet ja nopeudet annettiin taulukon 1 arvojen mukaisesti. Akselien kulmakiihtyvyyden arvoja ei taulukoissa ollut, joten niiden arvoksi jätettiin vakiona ollut  $2\pi \text{ rad/sec}^2$ . Kuviossa 10 on valmis pyörityspöydän malli. Runko-osasta on karsittu yksityiskohtia, mutta itse kiinnitysosa on mallinnettu tarkasti.

TAULUKKO 1. Akselien rajat ja nopeudet. (Fanuc 2007).

Item		Specifications	
Controlled axes		2 axes (J1, J2)	
Motion range	J1 axis	270° (-135° to +135°)	4.72 rad (-2.36 rad to +2.36 rad)
	J2 axis	480° (-240° to +240°)	8.38 rad (-4.19 rad to +4.19 rad)
Maximum speed	J1 axis	120°/sec	2.09 rad/sec
	J2 axis	190°/sec	3.32 rad/sec



KUVIO 10. Pyörityspöydän malli.

#### 6.4 Ympäristö ja oheislaitteet

Solumalliin lisättiin vielä muita laitteita joihin robotti ulottuu, kuten ympäröivät turva-aidat, työkalutelineet, langansyöttölaitteisto, hitsausvirtalähde ja jäähdytynyksikkö, hitsauspolttimen huoltoasema, levyn paikoitusasema, FMS-hissin verkkoseinä, toinen robotti sekä FMS-hissin konepalettien ja materiaalilavojen latausasemat. Nämä törmäystarkasteluun käytetyt laitteet sisällytettiin omaan mallitiedostoon, jonka väri määriteltiin valkoiseksi ja mallin ominaisuuksista estettiin sen valinta hiirellä, jolloin se ei tule vahingossa valituksi ohjelmia tehtäessä. Myös kalibrointiin käytetystä kalibrointityökalusta luotiin parametrinen malli, jolloin sen mitat voidaan antaa mittaustulosten perusteella.

## 6.5 CAD-mallien tuonti Delmia V5:een

Etäohjelmointisoluun voidaan tuoda Delmia V5:llä tehtyjen mallien lisäksi myös muilla ohjelmilla tehtyjä CAD-malleja. Autodesk Inventor -ohjelmasta saatiin onnistuneesti tuotua kokoonpanomalli STEP-muodossa, jolloin osien hierarkia säilyi. Tällöin kuitenkin osien muokkaaminen oli hankalaa. Tuotteen suunnittelun ja valmistuksen suunnittelun yhdistäminen vaatii, että tuote mallinnetaan samassa ohjelmassa.

## 7 ETÄOHJELMOINTISOLUN KALIBROINTI

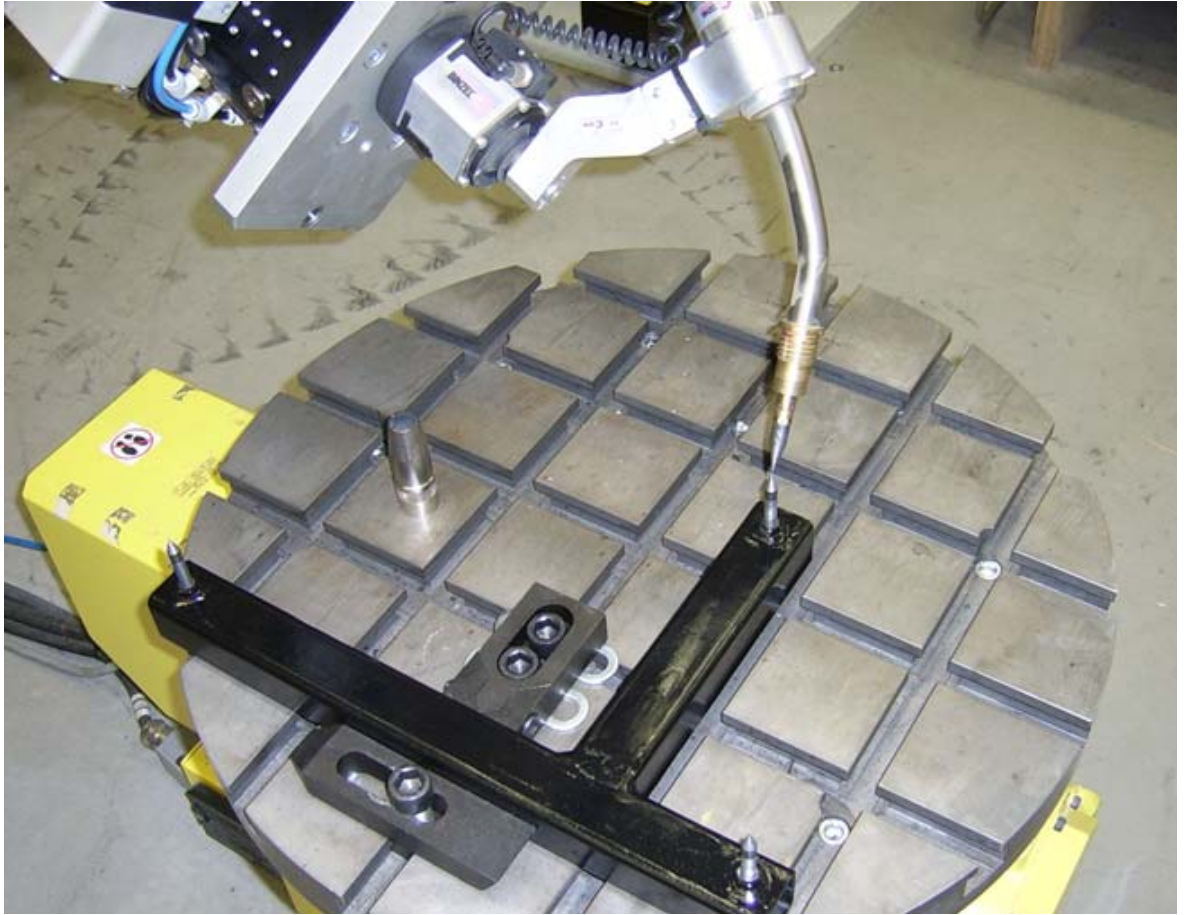
Etäohjelmointisolusta päätettiin kalibroida robotin nivelten nollakohdat, hitsauspolttimen työkalupiste, pyörityspöydän sijainti ja pöydän korkeus. Mittalaitteena päätettiin käyttää robottia ja tarvittavat mittavälineet valmistettiin kalibrointitarpeiden pohjalta. Ympäröivän solun mittaus päätettiin suorittaa rullamitalla.

### 7.1 Kalibroinnin mittavälineet

Mittausdatan keräämistä varten valmistettiin kaksi eripituista piikkiä, jotka sopivat robotin suuttimeen. Lyhyempi piikki sorvattiin 12 mm:n pituiseksi, jolloin se kuvastaa suoraan 12 mm:n pituisen vapaalangan päätä. Pidemmästä piikistä tehtiin noin 50 mm pitkä. Molempien piikkien kärkikulma oli 30 astetta.

Lisäksi valmistettiin kalibrointityökalu johon robottia voidaan kohdistaa. Kalibrointityökalun runko valmistettiin 40 x 40 mm:stä putkipalkista, jotta se olisi tukeva ja estäisi muodonmuutoksia. Runkoon liitettiin kolme terävää piikkiä, joiden kärjet toimivat vertailupisteinä mittauksia tehtäessä. Piikeistä tehtiin 50 mm pitkiä 30 asteen kärkikulmalla. Koska terävä kärki taipuu helposti, piikit sorvattiin lujuusluokan 12.9 M8-pulteista. Piikkien sijoittelulla ei ollut väliä, koska tarkoituksena oli saada kolme sattumanvaraista vertailupistettä. Kuviossa 11 näkyy kalibrointityökalu kiinnitettynä pyörityspöytään.





KUVIO 11. Kalibrintityökalu kiinnitettynä kalibrointia varten.

Jotta kalibrintityökalua voitiin käyttää vertailukohtana mittauksissa, oli sen piikkien kärkien etäisyydet tunnettava. Etäisyydet mitattiin Mitutoyon koordinaattimittalaitteella mittaamalla jokaisen piikin kartiolta neljä pistettä kahdelta eri tasolta. Mittalaitteen ohjelmistolla mitattiin kartioiden laskennallisten kärkipisteiden etäisyydet toisistaan, jolloin etäisyydet on kuvattavissa kolmella mitalla. Tämä oli yksinkertaisin tapa esittää kärkien paikat, koska ne ovat samalla tasolla. Jos piikkejä olisi ollut enemmän, olisi jokaista piikkiä kohden otettu ylös kärjen sijainti XYZ-koordinaatistossa.

## 7.2 Akselien nollakohtien ja työkalupisteen kalibrointi

Mittausdatan keräämiseksi kalibrointia varten kiinnitettiin lyhyempi piikki polttimeen ja kalibrintityökalu puristimella pyörityspöytään. Robotilla tehtiin uusi ohjelma pis-



teiden tallennusta varten. Polttimessa oleva piikki kohdistettiin viiteen eri asentoon kuhunkin kalibrintityökalun piikkiin. Jokainen piste tallennettiin ohjelmaan nivelarvoina. Mittausjärjestys kirjoitettiin ylös paperille, jotta myöhemmin voitiin kohdistaa kukin mittauspiste oikeaan vertailupisteeseen. Ohjelma ladattiin USB-tikulle ohjainkaapin liitännästä.

Koska piikkien kärjet eivät ole täysin terävät, kohdistettiin ne käyttäen kuvitteellisia kärkipisteitä. Tällöin niiden väliin jää pieni rako. Alla olevassa kuvassa näkyvät kärjet vastakkain kalibrintimittauksen aikana.



KUVIO 12. Piikkien kohdistus.

Kalibrintitoimintoa varten tehtiin tekstitiedosto, josta ohjelma lukee datan. Tiedoston ensimmäisellä rivillä annettiin vapausasteiden lukumääräksi kuusi. Pyörityspöytää ei liikutettu mittausten aikana. Toisella rivillä annettiin vertailuun käytettävä pisteryhmä ja lopuilla riveillä kunkin mittaustuloksen vertailupisteen nimi ja mittauslukemat. Vertailupisteryhmälle annettiin nimi RefTags. Vertailupisteet nimettiin Tag-etuliitteellä ja juoksevilla numerolla. Tämä on ohjelman käyttämä oletusnimeämistapa, jolloin pisteitä ei tarvitse erikseen nimetä ohjelmassa. Mittauslukemista kuusi ensimmäistä voitiin jättää nolliksi, koska niihin olisi asetettu ulkoisen koordinaattimittalaitteen lukemat, jos sellaista olisi käytetty. Kuuteen seuraavaan sarakkeeseen poimittiin ylösladatusista robottiohjelmasta nivelarvot. Tiedosto tallennettiin **dat** -päätteellä, jotta ohjelma tunnistaa sen mittausdataksi.

Mittaustuloksia syöttäessä desimaalierottimenä pitää käyttää pilkkua ja rivien loppuun lisätä välilyönti. Ohjelma on tarkka tiedoston rakenteesta ja se on syytä olla oikein. Kuviossa 13 on esimerkki datatiedoston alusta.

```
dofcount 6
taggroup RefTags
Tag1 0,0 0,0 0,0 0,0 0,000 0,000 0,000 25,017 26,442 -35,697 -1,399 -79,801 117,498
Tag2 0,0 0,0 0,0 0,0 0,000 0,000 0,000 38,612 3,610 -45,289 65,809 -41,935 44,974
Tag3 0,0 0,0 0,0 0,0 0,000 0,000 0,000 44,886 10,750 -61,095 103,560 -64,556 -21,075
```

KUVIO 13. Esimerkki kalibrointidatasta.

Kalibrointityökalun malli lisättiin solumalliin ja siihen luotiin tag -ryhmä samalla nimellä, joka kirjoitettiin mittausdata-tiedostoon. Ryhmää luotaessa valittiin **Modify Reference**, jolloin ryhmä tallentuu kalibrointityökalumallin tiedostoon. Tagryhmään lisättiin mittapisteitä vastaavat tag-pisteet piikkien kärkiin, huomioiden aiemmin ylös kirjattu mittausjärjestys. Näiden pisteiden orientaatioilla ei ole merkitystä, koska kalibrointityökalulla ei pystytä mittaamaan suuntia. Robotille luotiin kalibroitava työkalupiste nimellä CalibTool käyttämällä **New Tool Profile** -toimintoa. Työkalupisteen tyypiksi valittiin **On Robot**, koska työkalu on kiinnitetty nä robottiin.

Robotti asetettiin nivelten nollakohtien kalibrointitilaan valitsemalla **Set Signature Status** -toiminnolla **Joint Offsets**. Kalibrointi suoritettiin **Auto ID** -toiminnolla. Kalibroitavaksi laitteeksi valittiin robotti ja mittausdataksi edellä luotu tiedosto. Kalibroitaviksi akseleiksi valittiin robotin pääakselit ja kalibroitavaksi työkalupisteeksi CalibTool. Mittalaitteeksi valittiin piikkimatto ja vertailupisteiksi siihen sidottu tagryhmä. Kalibrointi parametreista asetettiin **Tag Orientation** -valintaan **No**, koska pisteiden todellisia orientaatioita ei ollut tiedossa. Mittausdatan kohinan arvioinnit annettiin olla oletusarvoina 0,1 mm ja 0,1°. Mittalaitteen siirtymät ja orientaatiot jätettiin vapaaksi valitsemalla kaikkiin **Free**. Työkalupisteen siirtymät jätettiin myös vapaiksi. Akseleiden vapausasteet lukittiin vielä tässä vaiheessa valitsemalla niille **Fixed**, koska ohjelma tuntui löytävän vaihtoehdoisen ratkaisun, jos piikkimatto sijaitsi liian kaukana todellisesta asemastaan. Kun kalibrointiparametrit hyväksyttiin, piikkimaton malli siirtyi lähelle oikeaa paikkaansa ja kalibrointitoimenpide voitiin suorittaa uudelleen. Tällä kertaa jätettiin myös akselit vapaiksi. Ohjelma ilmoitti kalibrointitulokset, joissa ilmeni mittausepävarmuudet, sovitettujen pisteiden luku-

määrä, mittapisteiden pienimmän neliösumman sovituservo (RMS) ja työkalupisteen siirtymät. RMS-arvo oli alle yhden, joten tulosta voitiin pitää hyvänä.

### 7.3 Työkalupisteen suunnan kalibrointi

Edellisen vaiheen perusteella tiedettiin jo työkalupisteen paikka lyhyellä piikillä, mutta ei sen orientaatiota. Orientaation selvittämiseksi polttimeen kiinnitettiin pitempi piikki ja mitattiin sillä kuusi pistettä kohdistamalla sitä pyörityspöytään ruuvattuun piikkiin. Vertailun vuoksi otettiin kuitenkin samat mittaukset myös lyhyellä piikillä. Pisteet tallennettiin uuteen ohjelmaan ja tällä kertaa pisteiden esitysmuodoksi valittiin nivelarvojen sijasta karteellinen koordinaatisto. Ennen pisteiden tallennusta robotin työkalupisteeksi asetettiin sellainen, jolla siirtymä- ja kiertymäarvot olivat nolliä.

Kalibrointipisteet sisältävä ohjelma tallennettiin USB-tikulle ja tuotiin simulointimaailmaan valitsemalla **Import Robot Program**. Kohteeksi valittiin robotti ja ladattavaksi ohjelmaksi tikulla oleva tiedosto. Kääntäjäksi valittiin Fanuc.jar. Robottiohjelma ja tag-pisteet ilmestyivät nyt simulointimaailmaan. Ohjelma voitiin poistaa, koska se oli tarpeeton ja vain tag-pisteitä tarvittiin.

Ladattu pisteryhmä jaettiin kahteen ryhmään, koska kalibrointitoiminto kysyy yhtä kokonaista tag -ryhmää, yhtä kalibroitavaa työkalupistettä kohden. Robotille luotiin myös kaksi uutta työkalupistettä lyhyelle (CalibShort) ja pitkälle (CalibLong) piikille.

Kalibrointi suoritettiin valitsemalla **Calibrate Tool Point** -toiminto. Kohdelaitteeksi valittiin robotti ja vertailupisteiksi lyhyen piikin tag-ryhmä. Kalibrointiparametreista valittiin kalibroitavaksi työkalupisteeksi CalibShort ja siirtymät jätettiin vapaiksi. Mittauskohinaksi annettiin 0,1 mm. Parametrit hyväksyttiin ja toimenpide toistettiin myös toiselle työkalupisteelle. Molempien RMS-arvo oli alle yhden.

Työkalupisteen orientaation määrittämiseksi ei ohjelmassa ole valmista toimintoa. Orientaatio laskettiin työkalupisteiden koordinaateista niin, että saatiin lyhyen piikin työkalupisteen Z-akseli kulkemaan pitkän piikin työkalupisteen läpi. Lisäksi Z-

akselin positiivinen suunta valittiin polttimeen päin. Työkalupisteen orientaatiolle löytyy kuitenkin ääretön määrä ratkaisuja, jos kaikkien akseleiden kiertymät jätetään vapaiksi. Niinpä tässä vaiheessa valittiin X-akselin suunta hitsauspolttimen kuljetussuuntaan päin, jolloin Y-akseli asettuu hitsattavaa kappaletta kohden.

#### 7.4 Pyörityspöydän kalibrointi

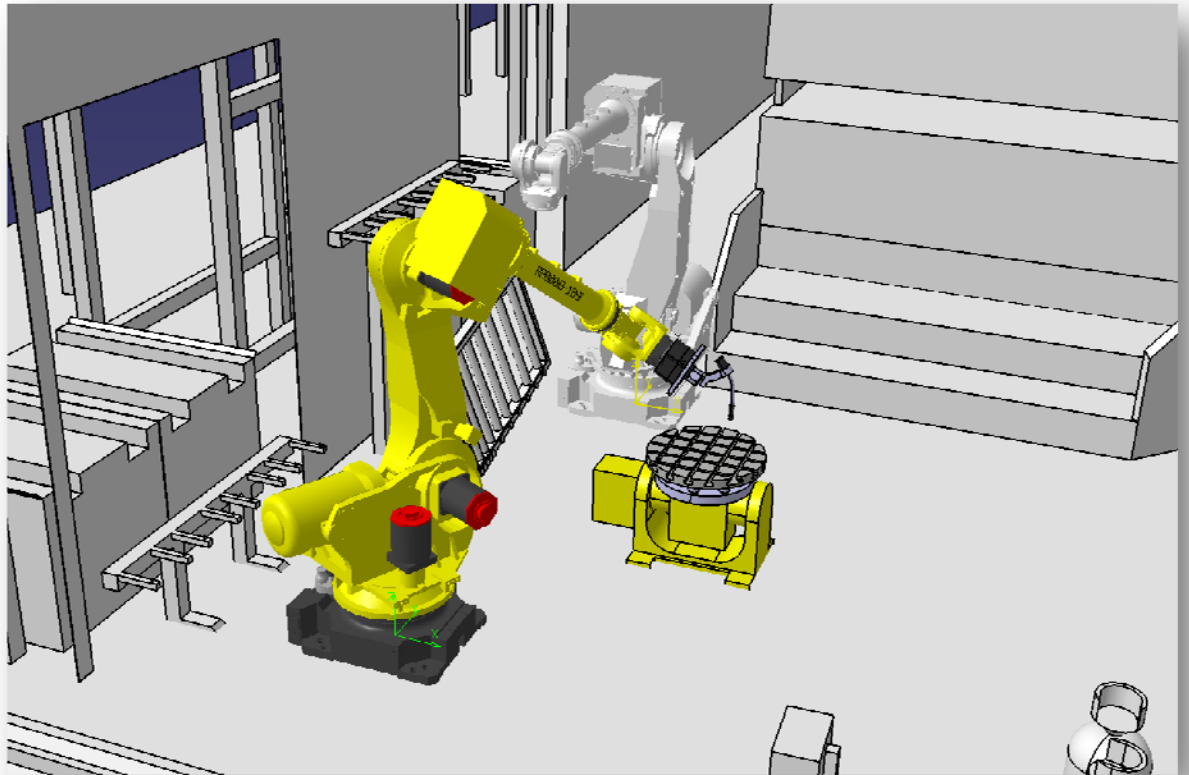
Pyörityspöydän tarkan sijainnin ja pöytäosan korkeuden selvittämiseksi kohdistettiin robotti pöydässä olevaan piikkiin niin, että pöytää käännettiin eri asentoihin. Pisteet sisältävä ohjelma ladattiin etäohjelmointimalliin, jossa pöytäosan origoon luotiin mittauspisteitä vastaavat pisteet. Kalibrointi suoritettiin **Positioner Calibration** -toiminnolla, jossa valittiin ensin robotti ja pyörityspöytä. Vertailupisteiksi valittiin pöytäosaan luodut pisteet ja mittauspisteiksi ladatut pisteet. Paikallaan pysyväksi osaksi valittiin robotin jalka. Kalibrointiparametreissa annettiin pöydän siirroille, orientaatioille ja työkalupisteen siirroille vapaudet. Mittauskohinaksi annettiin jälleen 0,1 mm. Nyt pöytä siirtyi oikeaan paikkaan ja ohjelma ilmoitti pöydän työkalupisteen eli siihen ruuvatus piikin kärjen paikan. Mittapiikin korkeus työntömitalla mitattuna pöydän pinnasta oli 22,5 mm. Simulointimallissa sama mitta oli 28,7 mm, joten pöydän pintaa oli korotettava 6,2 mm. Korotus tehtiin pöydän geometriaan kasvattamalla pöytäosaa tällä mitalla.

#### 7.5 Työkappaleen kalibrointi

Etäohjelmointisoluuun tuotujen työkappaleiden tarkkojen sijaintien selvittämiseksi mitattiin kiinnittimeen asetetusta kappaleesta robotilla kolme pistettä ja pisteet sisältävä robottiohjelma ladattiin etäohjelmointisoluuun. Työkappaleeseen tehtiin mittauspisteitä vastaava pisteryhmä. Kalibrointi suoritettiin käyttämällä **Calibrate Workpiece** -toimintoa, jolle annettiin ladatut pisteet ja kappaleeseen sidotut pisteet. Siirtymille ja kiertymille annettiin vapaudet joka suuntaan. Kappaleesta voitaisiin mitata useampiakin pisteitä tarkkuuden parantamiseksi, mutta kolmen pisteen mittaus todettiin riittäväksi käytetyille kappaleille. Myös kokoonpanon eri osien sijainti voitaisiin kalibroida erikseen.

## 7.6 Kalibroinnin tarkastus

Kalibroinnin onnistuminen tarkastettiin tekemällä robotille ohjelma, joka kohdistaa polttimessa olevan piikin pyörityspöydässä olevaan vastinpiikkiin useassa eri asennossa. Kärkien väliin jätettiin ohjelmoitaessa yhden millin rako, jotta piikkien kärjet eivät taipuisi kosketuksesta. Ohjelma ladattiin robotille ja tarkistuspisteet ajettiin läpi. Kohdistus arvioitiin silmämääräisesti ja kun suurempia poikkeamia ei esiintynyt, todettiin kalibrointi onnistuneeksi. Alla on kuvio kalibroidusta robotisolusta.



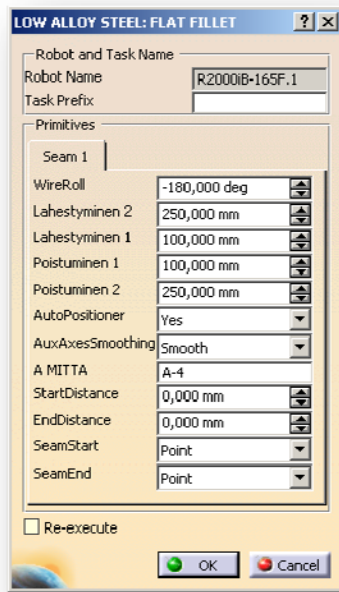
KUVIO 14. Kalibroitu robottisolu.

## 8 OHJELMOINTIYMPÄRISTÖN RÄÄTÄLÖINTI

Robotin ohjelmointi pyrittiin tekemään mahdollisimman yksinkertaiseksi ja nopeaksi. Tarkoituksena oli että hitsausohjelmia tehtäessä ei tarvitsisi tietää oikeita hitsausparametreja. Nykyisin hitsausradat luodaan AMP-hitsausmakroilla ja oikean hitsausparametritaulukon numero lisätään käsin ohjelmaan. Hitsausparametrien tallentamiseen ja hallintaan kehitettiin taulukkopohja Microsoft Excel -ohjelmalla. Hitsausparametrit sisältävä taulukkopohja integroitiin AMP-hitsausmakroiin VB Script -ohjelmalla, joka lisää parametritiedot ratojen luonnin yhteydessä. Parametrienhallintataulukko ja AMP-hitsausmakrot muodostavat yhdessä hitsauspiirteitä. Hitsauspiirre on tietyntyyppisen hitsin täydellinen kuvaus ja se on toistettavissa. Hitsauspiirteillä käyttäjä pystyy luomaan hitsausohjelmia valitsemalla hitsauskuvan mukaisen hitsityypin ja tarvittavan a-mitan. Piirrettä lisättäessä ohjelma käynnistää ensin hitsausmakron, joka laskee tarvittavan polun lähestymis- ja poistumispisteineen. Kun polku on luotu, ohjelma lisää siihen tarvittavat hitsausparametrit taulukosta.

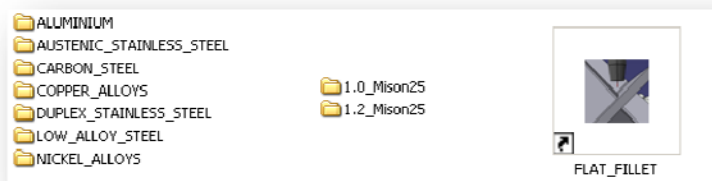
### 8.1 Hitsausmakrojen luonti

Delmia V5:ssä voidaan käyttää AMP-hitsausmakroja, joilla voidaan luoda hitsauspolkuja. Makroilla on mahdollista määritellä hitsausasento, poltinkulma, railonhakupisteet, lähestymis- ja poistumispisteet sekä monia muita hitsausmäärittelyjä. Yhteen hitsausmakroon voidaan myös määritellä useampi palko. Jotkin määreet voidaan jättää vapaiksi, jolloin käyttäjä pystyy muuttamaan niitä hitsauspolkuja luodessa. Kuvassa 15 on esimerkki hitsausmakron suorituksesta ohjelmassa. Ikkunassa näkyy myös lisätty a-mittakenttä, johon käyttäjä voi antaa halutun mitan.



KUVIO 15. Hitsausmakron suorittaminen.

Hitsausmakroista päätettiin tehdä tarkasti yksilöityjä tietyille hitsityypille ja hitsausasennolle. Hitsityypeistä ja hitsausasunnoista tulee jo melko monta variaatiota. Lisäksi kun huomioidaan, että jokaista materiaalityyppiä, käytettyä lisäainelankaa ja suojakaasua kohden on omat arvonsa, tulee hitsausmakroja suuri määrä. Variaatiot hitsattavan kappaleen paksuuksista jätettiin tässä työssä huomioimatta. Makrojen lajittelu tehtiin omiin kansioihin niin, että käyttäjä valitsee ensin hitsattavan materiaalin. Tämä kansion alta valitaan käytettävän lisäainelangan vahvuus ja suojakaasu. Kuvassa 16 vasemmalla on materiaalin valinta ja keskellä langan ja suojakaasun valinta. Tämä kansio asetetaan ohjelmaan kansioiksi, josta makroja luetaan. Asetus jää voimaan niin kauaksi, kunnes se muutetaan. Tämä kansio sisältää näille valinnoille soveltuvat hitsausmakrot. Kuvassa 16 oikealla on esimerkkinä jalkoasennossa hitsattavan pienahitsin makro.



KUVIO 16. Hitsausmakron valinta.

## 8.2 Hitsausparametrien hallinta

Hitsausparametrien hallinta toteutettiin Excel-taulukko-ohjelmalla. Jokainen makrokansio sisältää PARAMETERS.xls-tiedoston, jossa on jokaista kansion sisältämää hitsausmakroa kohden välilehti. Välilehdellä on eri a-mitoille käytettävän hitsaustaulukon numero ja vaaputuksen arvot. Kuvassa 17 on FLAT\_FILLET-makron parametrivälilehti. Kuljetusnopeutta ei tarvitse hitsausohjelmaan sisällyttää, koska se luetaan taulukosta ajon aikana. Se pitää kuitenkin ottaa huomioon etäohjelmoinnissa, jotta tahtiajoista saadaan todenmukaiset.

A_Value	Schedule number	Travel Speed	Weave	Description
1	Unset	Unset		Unset
2	Unset	Unset		Unset
3	20	45,0		Jalkosauma
4	21	40,0		Jalkosauma
5	22	35,0	Weave Sine[1.0Hz,1.5mm,0.200s,0.200s]	Jalkosauma
6	23	35,0	Weave Sine[1.0Hz,2.0mm,0.200s,0.200s]	Jalkosauma
7	Unset	Unset		Unset
8	Unset	Unset		Unset
9	Unset	Unset		Unset
10	Unset	Unset		Unset

KUVIO 17. FLAT\_FILLET-makron parametrivälilehti.

PARAMETERS.xls-tiedosto sisältää myös parametrien koontivälilehden, johon on koottu kaikki hitsaustaulukot ja niiden parametrit, joita sen sisällä on käytetty. Kuvassa 18 näkyy käytetyt taulukkonumerot 20–24. Hitsaustaulukoiden yläpuolelle on koottu eri taulukoissa käytettyjen prosessien numerot. Kuvassa näkyvissä taulukoissa on käytetty vain yhtä prosessinumeroa (6). Tämän yläpuolelle on koottu jokaisen prosessin käyttämän hitsausohjelman numero. Numero viittaa hitsauslaitteeseen tallennettuun hitsausohjelmaan, joka kuvan tapauksessa on *CV ArMix GMAW 1.0 mm Steel ArCO2*. Ylimpänä on lisäainelangan halkaisija ja suojakaasun tyyppi. Käyttäjä voi tulostaa tämän koontilistan ennen hitsausohjelman ajamista ja tarkastaa tarvittavat hitsauslaitteiston esiasetukset.

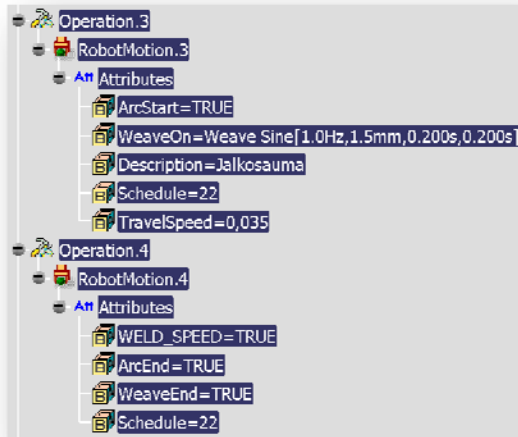


LOW ALLOY STEEL	Wire diameter	Gas						
	1.0	Mison 25						
Process number	Mode	Process description						
6	11	CV ArMix GMAW 1.0 mm Steel Ar CO2						
Schedule number	Process number	Process description	WFS	Volts/Trim	Pinch/Wave Control	Travel speed	Delay time	
20	6	CV ArMix GMAW 1.0 mm Steel Ar CO2	450	20	0	45	0	
21	6	CV ArMix GMAW 1.0 mm Steel Ar CO2	450	21	0	40	0	
22	6	CV ArMix GMAW 1.0 mm Steel Ar CO2	500	22	0	35	0	
23	6	CV ArMix GMAW 1.0 mm Steel Ar CO2	550	23	0	35	0	
24	6	CV ArMix GMAW 1.0 mm Steel Ar CO2	570	24	0	35	0	

KUVIO 18. Parametrien koontivälilehti.

### 8.3 Parametrien liittäminen hitsauspolkuun

Hitsausmakro luo vain tarvittavat hitsauspolun pisteet ja hitsausparametrit on lisättävä vielä erikseen liikekäskyjen alle. Tämä toteutettiin VB Script -makrolla, joka käy läpi hitsauspolun pisteet ja lukee tarvittavat parametrit oikealta parametrivälilehdeltä valitun a-mitan mukaisesti. Hitsauksen aloituspisteeseen lisätään tiedot hitsauksen aloituksesta, käytetystä hitsaustaulukosta ja vaaputuksesta. Hitsauksen loppupisteeseen lisätään tiedot hitsauksen ja vaaputuksen lopettamisesta sekä hitsaustaulukon numero. Kaikkiin pisteisiin, joissa hitsaus on päällä, lisätään vielä merkintä kuljetusnopeuden numeerisen arvon korvaamisesta WELD\_SPEED määreellä, jolloin robotti lukee hitsausnopeuden hitsaustaulukosta. Kuvassa 19 näkyy kuinka määriykset näkyvät robottiohjelman liikekäskyjen alla, kun ohjelmassa on suoritettu FLAT\_FILLET – hitsausmakro ja a-mitaksi valittu 5. Ohjelma poimii kuvassa 17 näkyvästä taulukosta hitsaustaulukon sekä vaaputuksen määrittelyn ja lisää ne oikeiden liikekäskyjen alle. Osa VB Script -makrosta löytyy tämän opinnäytetyön liitteestä.



KUVIO 19. Hitsausparametrien lisäys robottiohjelmaan liikekäskyjen alle.

#### 8.4 Robottikohtaisen kääntäjän muokkaus

Robottiohjelma tallentuu XML-tiedostoon, josta se voidaan kääntää eri robottikielille. XML-tiedosto sisältää tiedot liikekäskyistä ja niihin tallennetuista parametreista. Jotta ohjelma osaisi kääntää parametrit robottikieliseen listausmuotoon, on niiden käsittely määriteltävä robottikohtaisessa XLST-muotoisessa käännösohjeessa. Jotta parametreja voidaan käsitellä käännöksen aikana, ne on ensin luettava muuttujiin talteen. Muuttujat nimettiin samoilla nimillä kuin parametrit. Alla olevassa kuviossa näkyy kuinka ArcStart-parametri luetaan ArcStart-nimiseen muuttujaan. Myös muiden parametrien luku toteutettiin samalla tavalla.

```
<xsl:variable name="ArcStart" select="AttributeList/Attribute[AttributeName='ArcStart']/AttributeValue"/>
```

KUVIO 20. Parametrin lukeminen muuttujaan.

Hitsauksen aloituksen ja lopetuksen käsittely lisättiin kuvion 21 mukaisesti. Käännöksen aikana ohjelma testaa, onko ArcStart -muuttujan arvo TRUE. Jos ehto täyttyy, ohjelma lisää hitsauksen aloituskäskyn liikekäskyn perään ja lukee käytettävän hitsaustaulukon Schedule-muuttujasta. Ohjelma tekee saman myös hitsauksen lopetukselle.

```

<xsl:if test="$ArcStart = 'TRUE'">
  <xsl:text> Arc Start[</xsl:text><xsl:value-of select="$Schedule"/><xsl:text>]</xsl:text>
</xsl:if>
<xsl:if test="$ArcEnd = 'TRUE'">
  <xsl:text> Arc End[</xsl:text><xsl:value-of select="$Schedule"/><xsl:text>]</xsl:text>
</xsl:if>

```

KUVIO 21. Hitsauksen ja aloituksen käsittely.

Vaaputuksen aloituskäsky lisätään hitsauksen aloittavan liikekäskyn jälkeen seuraavalle riville. Koska vaaputuksen parametrit ovat saman liikekäskyn alla, ei kääntäjä lisää automaattisesti rivinvaihtoa, vaan se piti lisätä vaaputuksen käännösohjeeseen. Ennen rivinvaihtoa nykyinen rivi pitää ensin päättää puolipilkulla. Rivinvaihdon merkki luetaan käännöstiedoston alussa asetetusta cr-muuttujasta. Uudelle riville lisätään rivinumero ja se tasataan oikealle. Vaaputus-komento lisätään lukemalla se WeaveOn-muuttujasta. Vaaputuksen lopetus tehtiin samalla tavalla. Kuviossa 22 on vaaputuksen aloituksen määrittely käännösohjeessa. Kun ohjelma sitten käännetään robottikohtaiselle kielelle, hitsauksen aloitus- ja lopetuspisteet vaaputusmäärittelyineen näyttävät kuvion 23 mukaiselta.

```

<xsl:if test="$WeaveOn != ''">
  <xsl:text> ;</xsl:text>
  <xsl:value-of select="$cr"/>
  <xsl:variable name="linenum" select="NumberIncrement:next()"/>
  <xsl:call-template name="lineNumber">
    <xsl:with-param name="linenum" select="$linenum"/>
  </xsl:call-template>
  <xsl:text> </xsl:text><xsl:value-of select="$WeaveOn"/>
</xsl:if>

```

KUVIO 22. Vaaputuksen aloituksen määrittely.

```

5:L P[3] 100mm/sec FINE Arc Start[22] ;
6: Weave Sine[1.0Hz,1.5mm,0.200s,0.200s] ;
7:L P[4] WELD_SPEED FINE Arc End[22] ;
8: Weave End ;

```

KUVIO 23. Ohjelma Fanuc-robotille käännettynä.

## 9 OHJELMIEN YLÖS- JA ALASLATAUS

Ohjelmia ei pysty suoraan alaslataamaan robotille listausmuodossa, vaan ne on ensin muutettava binäärimuotoon. Myöskään binäärimuotoisia ohjelmia ei suoraan pysty ylöslataamaan Delmia V5:een, vaan ne on muutettava listausmuotoon. Binäärimuotoinen ohjelma voidaan tarvittaessa muuttaa listausmuotoon jo robotilla, mutta ei toisinpäin. Käännöksiin tarvitaan erilliset ohjelmat, jota ei toimiteta Delmia V5 -ohjelmiston mukana. Tarvittavat muunnosohjelmat saatiin Fanucin yhteyshenkilöltä. Muunnosohjelmapakettiin kuuluu useampi Windowsin komentokehotteessa ajettava ohjelma. Näistä tässä työssä käytettiin kolmea alla esiteltyä.

### 9.1 Setrobot

Setrobot -ohjelma tekee robottiohjainkohtaisen konfiguraatitiedoston, jota tarvitaan muunnosohjelman ohjeeksi. Ohjelmaan syötettiin tiedot robottiohjaimen ohjaamista laitteista. Akseliryhmien lukumääräksi annettiin kolme. Ensimmäinen akseliryhmä koostuu robotista, joksi valittiin 2000iA, koska 2000iB versiota ei siinä ollut. Tällä ei kuitenkaan havaittu mitään vaikutuksia myöhemmissä vaiheissa. Ulkoisten akselien ensimmäinen ryhmä on konepalettien latausaseman yhden akselin pyöritys. Ryhmän tyyppiä annettiin NOBOT, akselien lukumääräksi yksi ja akselin tyyppiä Rot. Toinen ulkoisten akselien ryhmä on kahden akselin pyörityspöytä, joten sen akselien lukumääräksi annettiin kaksi ja molempien akselien tyyppiä myös Rot. Ohjelma tekee robot.cfg nimisen tiedoston kansioon, jossa ohjelma sijaitsee.

## 9.2 Maketp

Maketp-ohjelma muuntaa listausmuotoisen robottiohjelman binäärimuotoon noudattaen edellä luodun konfiguraatitiedoston määrittelyksiä. Konfiguraatitiedoston polku, käännettävän ohjelman polku ja kohdetiedoston polku annetaan komentokehoteessa parametreina ohjelmalle. Kohdetiedoston nimen piti olla sama, joka oli annettu ohjelmatiedoston määrittelyosiossa. Jos ohjelmatiedoston nimeä halusi vielä tässä vaiheessa muuttaa, muutos piti tehdä sekä tiedoston nimeen, että määrittelyihin. Ohjelma keskeytti käynnöksen, jos käännettävässä ohjelmassa tai konfiguraatitiedostossa oli virheitä. Testausvaiheessa huomattiin kuitenkin, että virheellisiä ohjelmia pystyi viemään robotille asti, jolloin robotin ohjainyksikkö meni jumiin ja tarvitsi uudelleenkäynnistyksen toimiakseen taas. Tällainen ongelma syntyi, jos ohjelmatiedoston määrittelyosiossa oli määritelty käytettäväksi useampia akseleita, mutta paikkatiedoissa ei oltu annettu arvoa näille akseleille.

## 9.3 Printtp

Printtp:llä voidaan tulostaa binäärimuotoinen robottiohjelma listausmuotoon. Tuloksen voi tarkastaa suoraan komentokehoteesta tai sen voi tallentaa tekstitiedostoon. Kun ohjelmia vietiin robotille, tarkastettiin binäärikäännöksen onnistuminen kääntämällä ohjelma takaisin listausmuotoon. Lopullisen etäohjelmointimallin ohjelmien kääntämisessä ei enää havaittu virheitä, mutta niitä voi syntyä, jos esimerkiksi XLST-käännösohjeeseen tehdään muutoksia tai lisäyksiä. Binäärimuotoista ohjelmaa ei pysty suoraan lataamaan etäohjelmointiympäristöön, vaan se on muutettava listausmuotoon.

## 9.4 Käännöksen automatisointi

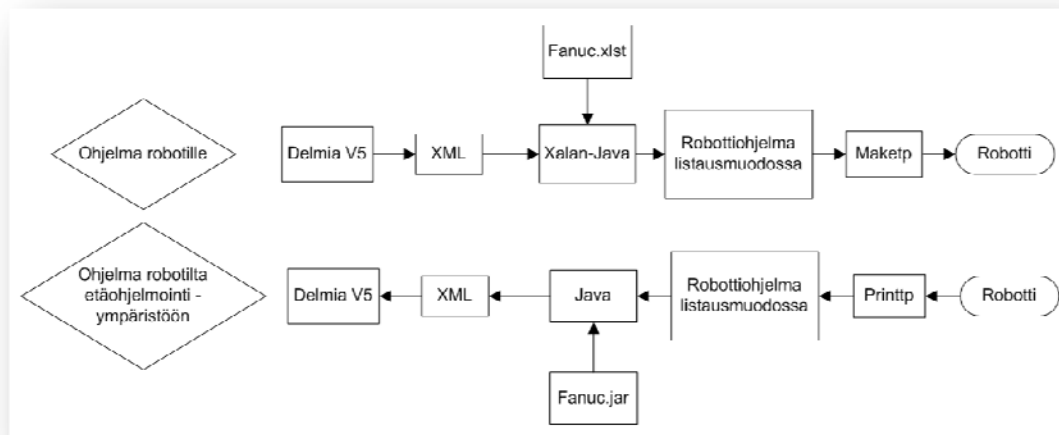
Parametrien syöttämiseen komentokehoteeseen menee paljon aikaa, joten siihen päätettiin tehdä nopeuttava työkalu. Yksinkertaisin tapa oli käyttää Batch-ohjelmointia. Ohjelma käy läpi kaikki robottiohjelmat määrätystä polusta ja suorittaa niille käännöksen Maketp:n avulla. Ohjelma siirtää käännettyt robottiohjelmat

KAANNETYT-kansioon. Lisäksi ohjelma tekee jokaiselle käännökselle takaisin-käännön Printtp:n avulla ja avaa ne Notepad-ohjelmaan tarkastettavaksi. Batch-ohjelman listaus on tämän opinnäytetyön liitteenä.

## 9.5 Ylös- ja alaslatauksen vaiheet

Robottiohjelman alaslatauksessa Delmia V5 muodostaa ensin XML-tiedoston valitusta ohjelmasta. XML-tiedosto käännetään robottikohtaiselle kielelle XSLT-tiedoston ohjeiden mukaisesti. Tämä listausmuotoinen ohjelma muunnetaan binäärimuotoon Maketp:n avulla.

Delmia V5:een ohjelmia ylösladattaessa ohjelma tallennetaan robotilta joko suoraan listausmuotoon tai käytetään Printtp:tä. Tämä ohjelma annetaan robottikohtaiselle Java-parserille, joka jäsentää ohjelman XML-muotoon. Lopuksi Delmia V5 muodostaa XML-tiedostosta ohjelmassa näkyvän hierarkiapuun. Kuviossa 24 on hahmotettu edellä mainitut ohjelmien ylös- ja alaslatauksessa tapahtuvat vaiheet.



KUVIO 24. Robottiohjelmien ylös- ja alaslatauksen vaiheet.

## 10CASE SLV

Seinäjoen Ammattikorkeakoululla tehtiin menetelmätestausta Seinäjoen Laatuvaruste Oy:lle liittyen pyöreän holkin robottihitsaamisesta ruostumattomaan pyöreään putkeen. Hitsauksesta oli tehty Delmian Igrip -ohjelmalla hitsausrata ja tarkoituksena oli selvittää oikeat hitsausparametrit. Sauma oli kuitenkin epäsymmetrinen eikä sitä saatu korjattua parametreilla. Polkua pystyttiin siirtämään pystysuunnassa ja huomattiin, että jo puolen millin ero hitsaussaumassa synnytti lähes täysin erinäköisen sauman. Sauman muoto korjaantui, kun polttimen työkalupiste oli saatu kalibroitu. Vanhalla työkalupisteellä poltin oli noin 3 astetta vinossa, joka vielä kertaantui siirryttäessä hitsattaessa holkin eri puolilla.

Oikeita hitsausparametreja haettiin valitsemalla aluksi järkevältä tuntuvat alkuarvot, joita sitten muutettiin yksi kerrallaan. Muuttuvia parametreja olivat kuljetusnopeus, langan syöttönopeus, kuljetuskulma, kallistuskulma sekä vapaalangan pituus ja jännite. Muuttumattomia parametreja olivat kaasutyyppi sekä lisäainelangan tyyppi ja paksuus. Lisäksi esiintyi laitteistosta johtuvia hitsiin vaikuttavia seikkoja, joihin ei pystytty vaikuttamaan, kuten syöttökaapelin kiertyminen hitsauksen aikana ja sen vaikutus langansyöttöön. Muuttuvia parametreja oli paljon ja sauman eri osissa parametrit olivat erilaiset. Testihitsauksia tehtiin noin 50 riittävän hyvien parametrien löytämiseksi.

## 11 TULOKSET JA KEHITYSKOhteet

Työni pääasiallisena tavoitteena oli, että malli saataisiin niin tarkaksi, että sillä tehtyjä ohjelmia voidaan ajaa suoraan robotilla. Tarkastusmittausten perusteella tähän tavoitteeseen päästiin. Hitsauspolttimen kalibrointi parantaa myös robotinohjaimesta tehtäviä hitsausohjelmia, kun polttimen orientaatiot voidaan antaa tarkasti numeroarvoilla.

Yhtenä työn tavoitteena oli helpottaa ja nopeuttaa hitsausohjelmien tekemistä. Tähän tavoitteeseen pääsyä rajoitti hieman se, että AMP-hitsausmakroilla tehtyjen hitsausratojen parametreja ei voi muuttaa sen jälkeen, kun polku on kerran luotu. Polkua voi muuttaa jälkeinpäin piste kerrallaan, mutta parametreja säätämällä se olisi helpompaa ja nopeampaa. Nämä puutteet eivät ole kuitenkaan merkittäviä, kun käyttäjä on tarkka parametreja asettaessa.

Tehtyä etäohjelmointimallia voidaan tulevaisuudessa laajentaa lisäämällä siihen toinen FMS-solussa sijaitseva robotti. Robottia voidaan hyödyntää kappaleenkäsittelijänä, jolloin monimutkaistenkin hitsien toteutus on mahdollista. Myös yhteydet FMS-järjestelmään voidaan rakentaa etäohjelmointimalliin, koska yhteyden muodostus tapahtuu robottiohjelmasta.



## 12 YHTEENVETO JA POHDINTOJA

Robottihitsauksen etäohjelmointi on jonkin verran jäljessä verrattuna koneistuskeskusten ja sorvien etäohjelmointiin käytetyistä CAM-ohjelmista. Tällä työllä on lähestytty näiden ohjelmien ominaisuuksia ja parannettu robottihitsauksen etäohjelmoinnin käytettävyyttä. Tällä on merkitystä erityisesti piensarjojen hitsaukseen robotilla, koska ohjelmointiaikaa voidaan lyhentää.

Delmia V5:n hyödyntämisestä etäohjelmointiin saatiin arvokasta tietoa. Ohjelmassa on havaittavissa huomattavaa potentiaalia monimutkaistenkin järjestelmien etäohjelmointiin. Yksi suurimmista eduista on, että CAD-mallit voidaan tehdä samalla ohjelmalla. Pienten muutosten tekeminen malleihin on täten nopeaa. Ohjelmassa oli kuitenkin havaittavissa, että Delmian tekemä robottiohjelmointilaajennus CATIA V5:n päälle ei ollut vielä täysin yhteensopiva. Tämä ilmeni niin, että laajennuksen sisältämiä osia ei voinut käyttää makro-ohjelmoinnissa yhtä laajasti kuin ohjelman perustoimintoja. Delmia V5:n manuaalissa mainitaan, että AMP-työkalua ollaan mahdollisesti muuttamassa avoimemmaksi kehittäjiä varten (Online Documentation 2009). Tämä mahdollistaisi sen monipuolisemman käytön. Työn aikana testattiin, että VB Script -ohjelmalla oli helppo lukea tuotemalliin liitetyt hitsausmerkinnät. Hitsausmerkintöjen perusteella ohjelmalla voitaisiin tehdä hitsausohjelma automaattisesti. Ainoa este automatisoinnille oli, että AMP-hitsausmakroja ei pysty käynnistämään VB Script -ohjelmalla.

Hitsauspiirteet ovat yksilöllisiä tietyille hitsityypille ja käytetylle laitteistolle, joten ne pitää luoda sovelluskohtaisesti. Uutta etäohjelmointijärjestelmää valmistettaessa on hyvä selvittää yleisimmät hitsityypit joita robotilla tullaan hitsaamaan, koska jokaiselle hitsityypille ei ole mielekästä tehdä valmiiksi hitsauspiirteitä. Hitsauspiirteitä voidaan lisätä järjestelmään käyttöönoton jälkeen kun uudelle hitsauspiirteelle havaitaan tarve. Järjestelmää käyttävässä yrityksessä on hyvä olla yksi tai useampi pääkäyttäjä joilla on menetelmätietoa hitsauksesta. He voivat tarvittaessa päivittää ja lisätä hitsauspiirteitä muiden käyttäjien keskittyessä ohjelmointiin.

Työssä kehitetyn etäohjelmoinnin toimintamallin avulla Seinäjoen ammattikorkeakoulun tekniikan yksikkö pystyy tarjoamaan sekä korkeatasoista että ajanmukaista etäohjelmoinnin opetusta ja siihen liittyvää palvelutoimintaa. Etäohjelmoinnilla tehtyjä ohjelmia on helppo käydä ajamassa oikealla robotilla, koska se sijaitsee samassa rakennuksessa. Etäohjelmointia voidaan esitellä ammattikorkeakoululla käyville vierailijoille ja yritysedustajille. Näin kiinnostusta etäohjelmointia kohtaan voidaan kasvattaa ja luoda pohjaa tähän liittyvälle palvelutoiminnalle.

## LÄHTEET

2-axes Servo Positioner Maintenance Manual. 2007. [CD\_ROM].  
Fanuc.

Aalto, H. 2005. IGRIP UltraArc. [WWW-dokumentti]. Delfoi Oy. [Viitattu 1.4.2010]. Saatavissa:  
[http://www.delfoi.com/web/solutions/production/robotiikka/fi\\_FI/off-line/\\_files/73872456328021977/default/UltraArc\\_lyhyt%20esitys3.pdf](http://www.delfoi.com/web/solutions/production/robotiikka/fi_FI/off-line/_files/73872456328021977/default/UltraArc_lyhyt%20esitys3.pdf)

Aalto, H., Heilala, J., Hirvelä, T., Kuivanen, R., Laitinen, M., Lehtinen, H., Lempiäinen, J., Lylynoja, J., Renfors, J., Selin, K., Siintoharju, T., Temmes, J., Tuovila, T., Veikkolainen, M., Vihinen, J. & Virtanen, A. 1999. Robotiikka. Vantaa: Talentum Oyj / Metallitekniikka

Aaltonen, K. & Torvinen, S. 1997. Konepaja-automaatio. Porvoo: WSOY.

Bölmsjö, G., Loureiro, A. & Pires, J. N. 2006. Welding Robots. Lontoo: Springer.

GMAW Welding Guide. 2006. [WWW-dokumentti]. Lincoln [Viitattu 27.3.2010]. Saatavissa:  
<http://content.lincolnelectric.com/pdfs/products/literature/c4200.pdf>

Online Documentation. 2009. [CD\_ROM]. Dassault Systems.

Teollisuusrobottitilasto 2007. 2008. [WWW-dokumentti]. Suomen Robotiikkayhdistys ry. [Viitattu 12.1.2010]. Saatavissa:  
[http://www.roboyhd.fi/index.php?option=com\\_docman&task=doc\\_download&gid=31&Itemid=66](http://www.roboyhd.fi/index.php?option=com_docman&task=doc_download&gid=31&Itemid=66)

Teollisuusrobottitilasto 2008. 2009. [WWW-dokumentti]. Suomen Robotiikkayhdistys ry. [Viitattu 12.1.2010]. Saatavissa:  
[http://www.roboyhd.fi/index.php?option=com\\_docman&task=doc\\_download&gid=32&Itemid=66](http://www.roboyhd.fi/index.php?option=com_docman&task=doc_download&gid=32&Itemid=66)

## LIITE 1: Batch -ohjelma

```
1 @echo off
2 setlocal EnableDelayedExpansion
3
4 set MAIN_PATH=G:\OP\FANUC
5 set CONFIG_PATH=%MAIN_PATH%\robot.cfg
6 set PROG_PATH=%MAIN_PATH%\Ohjelmia\
7 set TRANSLATED_PROG_PATH=%MAIN_PATH%\Ohjelmia\KAANNETYT
8
9 cd %PROG_PATH%
10 for %%f in (*.ls) do (
11
12 set LS_NAME=%%f
13 set BASE_NAME=!LS_NAME:.ls=%!
14 set TP_NAME=!BASE_NAME!.TP
15 set TPPRINT_NAME=!BASE_NAME!.PE
16
17 %MAIN_PATH%\maketp.exe -C %CONFIG_PATH% !LS_NAME! !TP_NAME!
18 %MAIN_PATH%\printtp.exe -C %CONFIG_PATH% !TP_NAME! !TPPRINT_NAME!
19 notepad !TPPRINT_NAME!
20 del !TPPRINT_NAME!
21 move !LS_NAME! %TRANSLATED_PROG_PATH%
22 )
```

## LIITE 2: Hitsausmakroesimerkki

```
HEADER "LOW ALLOY STEEL: FLAT"  
  --COMMENT "Anna hitsaus polun parametrit"  
DEFAULTS  
END DEFAULTS  
  
SEAM  
  WeldSpeed FIXED 100.0  
  ApproachDirection FIXED "Out"  
  
  BaseTangentOffset FIXED 11  
  WallTangentOffset FIXED 10  
  
  WireRoll FREE -180  
  WireAlignment FIXED "Bisector"  
  WireSeamGravity FIXED "Z-X"  
  --RakeAngle FREE 0 LABEL "KULJETUS KULMA"  
  Alignment FIXED "Seam"  
  SeamAlignment FIXED "With Seam/Gravity"  
  SeamName FIXED "SAUMA"  
  
  --LAHESTYMINEN  
  ViaSpeed FIXED 300.0  
  SafeApproachDistance FREE 250 LABEL "Lahestyminen 2"  
  ApproachDistance FREE 100 LABEL "Lahestyminen 1"  
  
  --Poistuminen  
  DepartureDistance FREE 100 LABEL "Poistuminen 1"  
  SafeDepartureDistance FREE 250 LABEL "Poistuminen 2"  
  
  --Pyörityspöytä  
  AutoPositioner FREE "Yes"  
  AuxAxesSmoothing FREE "Smooth"  
  --AutoPositionerDirection FIXED {0 0 -1}  
  
  ScriptVar FREE "A-4" LABEL "A MITTA"  
  --Sauma  
  StartDistance FREE 0  
  EndDistance FREE 0  
  
  StartFlareAngle FIXED 0  
  EndFlareAngle FIXED 0  
  StartFlareLength FIXED 0  
  EndFlareLength FIXED 0  
  
  BaseSurface PROMPT "Valitse pohja"  
  WallSurface PROMPT "Valitse seinämä"  
  SeamStart FREE "Point"  
  SeamEnd FREE "Point"  
  
  Execute "C:\AMP\PRIMITIVES\LOW_ALLOY_STEEL\1.0_Mison25\PARAMETERS\FLAT.CATScript"  
END SEAM  
END
```

### LIITE 3: VB Script -esimerkki

```
40 Dim objAMPTags() As AMPTag
41 Dim count As Integer
42 iAMPPath.GetNumTags count
43 ReDim objAMPTags(count)
44 iAMPPath.GetAllTags objAMPTags
45 Dim objAMPTag As AMPTag
46 Dim tagType As String
47 Dim weldon as Boolean
48 Dim weaveon as Boolean
49
50 Dim i as Integer
51
52 For i=0 To count - 1
53
54     Set objAMPTag = objAMPTags(i)
55     tagType = ""
56     objAMPTag.GetType tagType
57
58     If weldon = True Then
59         objAMPTag.SetWeldSpeed TravelSpeed
60         objAMPTag.AddAttributeString "WELD_SPEED", "TRUE"
61     End If
62
63     If tagType = "Start" Then
64
65         weldon = True
66         objAMPTag.AddAttributeString "ArcStart", "TRUE"
67         objAMPTag.AddAttributeReal "Schedule", Schedule
68         objAMPTag.AddAttributeReal "TravelSpeed", TravelSpeed
69         If Weave <> "" Then
70             objAMPTag.AddAttributeString "WeaveOn", Weave
71             weaveon = True
72         End If
73         objAMPTag.AddAttributeString "Description", Description
74         objAMPTag.RemoveAttribute "Gun"
75
76     ElseIf tagType = "End" Then
77         weldon = False
78         objAMPTag.AddAttributeString "ArcEnd", "TRUE"
79         objAMPTag.AddAttributeReal "Schedule", Schedule
80         If weaveon Then
81             objAMPTag.AddAttributeString "WeaveEnd", "TRUE"
82             weaveon = False
83         End If
84         objAMPTag.RemoveAttribute "Gun"
85     End If
86
87
88
89 Next
```