INDUSTRIAL ROBOT CELL'S START-UP OPTIMIZATION BY UTILIZING ROBOTIC SIMULATION

Lappeenranta–Lahti University of Technology LUT

Master's Programme in Mechatronic System Design, Master's thesis

2022

Samu Peura

Examiners: Professor Aki Mikkola

D.Sc. (Tech.) Kimmo Kerkkänen

Advisor: M.Sc. (Tech.) Jarkko Brunou

ABSTRACT

Lappeenranta–Lahti University of Technology LUT

LUT School of Energy Systems

Mechanical Engineering

Samu Peura

Industrial robot cell's start-up optimization by utilizing robotic simulation

Master's thesis

2022

104 pages, 38 figures, 30 tables and 4 appendices

Examiners: Professor Aki Mikkola and D.Sc. (Tech.) Kimmo Kerkkänen

Advisor: M.Sc. (Tech.) Jarkko Brunou

Keywords: robot commissioning, optimization, parametrization, robotic simulation

The objective of this master's thesis was to optimize the commissioning of the Smelt Spout Robot because it was time-consuming and required a high level of robotics expertise. Due to the growing order backlog, proactive measures had to be taken. Therefore, the robotic simulation was used to expedite and facilitate the commissioning.

The literature review was focusing on the robot programming by introducing the mathematical background of the motion programming and the basics of KUKA Robot Language. The research continued by introducing the features of the KUKA Sim 4.0 simulation software, that was used in the case where the purpose was to optimize and parametrize the robot programs in the virtual world. They were implemented by creating a new teaching process for the coordinate systems and utilizing the accurate measuring tools of the robotic simulation and new features of the programming language. The development steps were tested with a real robot in the practical tests.

The results showed that the robotic simulation was an efficient and accurate tool to parametrize the robot programs. However, the practical tests showed that the manufacturing inaccuracies should be considered in the real world by the control parameters. In any case, the benefits of the simulation were significant, so its use will increase in the future. With the development steps of the research, the commissioning was significantly accelerated.

TIIVISTELMÄ

Lappeenrannan–Lahden teknillinen yliopisto LUT

LUT Energiajärjestelmät

Konetekniikka

Samu Peura

Teollisen robottisolun käyttöönoton optimointi robottisimulaation avulla

Konetekniikan diplomityö

104 sivua, 38 kuvaa, 30 taulukkoa ja 4 liitettä

Tarkastajat: Professori Aki Mikkola ja TkT Kimmo Kerkkänen

Ohjaaja: DI Jarkko Brunou

Avainsanat: robotin käyttöönotto, optimointi, parametrisointi, robottisimulaatio

Diplomityön tavoitteena oli optimoida sulakoururobotin käyttöönotto, koska se oli aikaa vievää ja vaati korkeaa osaamista robotiikasta. Kasvavan tilauskannan vuoksi, ennakoivia toimenpiteitä oli tehtävä. Tästä syystä robottisimulaatiota käytettiin nopeuttamaan ja helpottamaan käyttöönottoa.

Kirjallisuuskatsauksessa perehdyttiin tarkemmin robotin ohjelmointiin, esittelemällä liikkeiden ohjelmoinnin matemaattinen tausta ja KUKA Robot Language ohjelmointikielen perusteet. Tutkimus jatkui KUKA Sim 4.0 robottisimulaation ominaisuuksien esittelyllä. Simulaatiosovellusta hyödynnettiin tapauskohtaisessa tutkimuksessa, jossa tarkoitus oli optimoida ja parametrisoida sulakoururobotin ohjelmat virtuaalisessa maailmassa. Tavoitteet toteutettiin luomalla uusi opetusmenetelmä koordinaatistoille ja hyödyntämällä robottisimulaation tarkkoja mittaustyökaluja ja ohjelmointikielen käyttämättömiä ominaisuuksia. Kehitysaskeleet testattiin oikealla robotilla käytännön testeissä.

Tutkimuksen tulokset osoittivat, että robottisimulaatio oli tehokas ja tarkka työkalu robottiohjelmien parametrisointiin. Käytännön testit osoittivat, että reaalimaailmassa esiintyviä valmistusepätarkkuuksia tulee korjata säätöparametreilla. Simulaation tarjoamat edut olivat joka tapauksessa niin merkittävät, että sen hyödyntäminen tulee lisääntymään tulevaisuudessa. Tutkimuksen kehitysaskeleiden myötä, käyttöönottoa saatiin nopeutettua ja helpotettua merkittävästi.

ACKNOWLEDGEMENTS

I am grateful to the Recovery Boiler Smart Business Unit of ANDRITZ Oy, which provided a very interesting topic for this master’s thesis. Especially, I would like to thank my advisor Jarkko Brunou, who provided important advice during the research. I also thank Antti Hulkkonen from SoftMec Oy, whose robot expertise enabled the automation of new development stages.

I also thank Professor Aki Mikkola and D. Sc. (Tech.) Kimmo Kerkkänen from LUT University, who supervised me in this research. It was great to have you as the examiners of the research. With your suggestions for improvement, the research became higher quality and clearer.

An insanely big thanks to my beloved M. Sc. Johanna Kallinen. Thanks to her mathematical abilities and teaching sessions of the mathematical theories, I was able to approach research from a more theoretical perspective. Thanks also to friends and family who have supported me during the research.

Finally, I want to thank my friends from LUT University who have made the study time memorable. It has been awesome to study with you!

Samu Peura

Prague, 7.2.2022

SYMBOLS AND ABBREVIATIONS

API	Application Programming Interface
AR	Augmented Reality
CAD	Computational
CIRC	Circular motion type
DOF	Degrees of Freedom
DT	Digital Twin
FAT	Factory Acceptance Testing
IFR	International Federation of Robotics
I/O	Input/Output
KRL	KUKA Robot Programming
KRP	Recover & Power
KSS	KUKA System Software
LIN	Linear Motion
OLP	Offline Programming
PLC	Programmable Logic Controller
PNP	Plug and Play
PTP	Point-to-Point motion type
RB	Recovery Boiler
REL	Relative motion
TCP	Tool Center Point
VC	Virtual Commissioning
VR	Virtual Reality

Table of contents

Appendices

1. Introduction

The industry is undergoing a transformation where robots and automation are replacing a human routine work. They are facilitating production lines, increasing personal safety, and reducing costs. Due to the increasing use of the robots, the need for robot skills increases as the number of traditional jobs decrease. Regardless of a development of technology, the robot programming is still quite laborious and there is a shortage of experts.

As the need for robots grows at a rapid pace, it is important to focus on improving and speeding up current habits. As a result, simulation software has entered the market in recent decades that allows to plan the movements of the robot in advance in the simulation environment. In addition, simulation software can be used to improve old programs by parametrization. Therefore, the usability of the simulation is considered in this research relating to the commissioning of a Smelt Spout Robot (Figure 1).

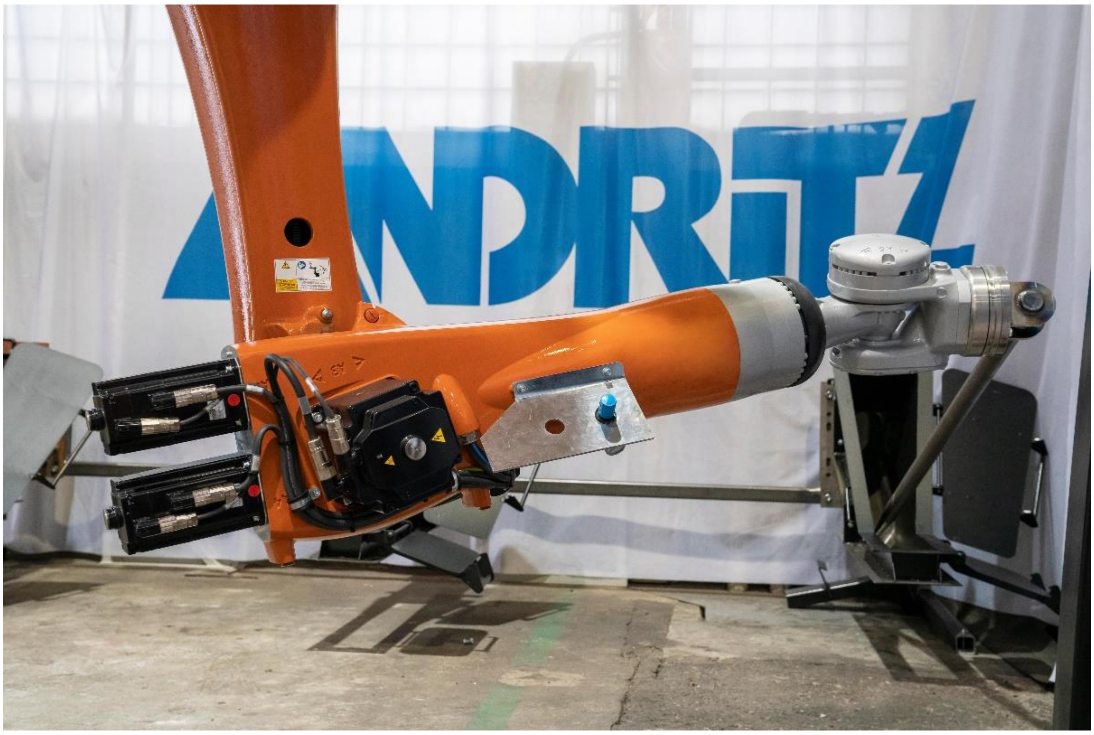


Figure 1. The Smelt Spout Robot at the testing environment in ANDRITZ Warkaus Works in Varkaus, Finland (Andritz 2019).

1.1 Motivation and background of the research

The first steps towards industrial robots can be considered started already in the year 1938 when the first programmable mechanism was developed, even if the first actual industrial robots were developed in the 1950s (Gasparetto & Scalera 2018, p. 286). According to the Gasparetto & Scalera (2018, p. 284), the development path of the industrial robots can be roughly divided into four different generations: “first generation (1950–1967), second generation (1968–1977), third generation (1978–1999), fourth generation (from 2000 on)”. From a general point of view, the current generation of robots is part of a new industrial revolution that is also known as Industry 4.0 which is considered have begun at the beginning of the millennium (Goel & Gupta 2019, p. 159). The previous generations can be included in the previous industrial revolution where “the robot boom” can be said to have begun in the 1980s (Gasparetto & Scalera 2019, p. 291).

With the new revolution of industry, the use of industrial robots has increased significantly. At the same time, a traditional robot programming is ineffective. (Pan et. all 2012, p. 88.) Due to the growing demand, the problem areas such as the robot programming should be intensified because the robot program that is built in the traditional style can also affect the commission of the robot, making it extremely slow. One solution could be the robotic simulations which utilizations have increased due to the Industry 4.0 (de Paula Ferreira 2020, p. 1). Thus, many robot manufacturers have developed the simulation software for the customers to simulate the movements of the robots.

The robotic simulation has many advantages. It allows the robot cell can be optimized during the design phase. There the robot programs can be visualized to other engineers using a simulation environment. (Scheidegger et al. 2018, p. 475.) It is also efficient programming tool because in the simulation, the shapes of the 3D models can be utilized in the development of the program. Thus, new efficient and parameterized program can be created to speed up the commissioning. (Pan et. all 2012, p. 90.) The robotic simulation has also become an efficient way to test the performance of the robot, develop new concepts, evaluate risks, select a robot, and reduce the costs of the project (de Paula Ferreira 2020, p. 2).

Due to the benefits of the simulation, it is used in this research to optimize the commissioning of the Smelt Spout Robot. The current programs of the Smelt Spout Robot are made by utilizing the traditional programming methods. There the robot should be manipulated by the commissioner; thus, the commissioning of the robot requires a lot of robotic expertise. The levels of competence in robotics are roughly divided into three stages in this research:

1. High level of competence – A robot expert
2. Medium level of competence – Some knowledge about robotics
3. Low level of competence – Not specific knowledge about robotics.

The robot expert can program the robots and has high knowledge of robotics by working with them for a long time. The person on the second level knows the basics of robotics and can understand the directions of the coordinate systems, and therefore can modify the parameters in the robot program. The person on the lowest level does not have much knowledge of robotics, but he can modify the parameters in a panel according to the instructions.

The Smelt Spout Cleaning System is carried out by utilizing the latest robotic technology from the German industrial robot manufacturer KUKA. The Smelt Spout Robots are located to a smelt spout area of a recovery boiler in front of the smelt spouts. The main purpose of the robotic arm is to clean up three ANDRITZ standard-spaced smelt spouts using a hi-tech cleaning rod. The system has been developed to decrease working in an environment that is potentially dangerous to humans.

The delivery of the Smelt Spout Cleaning System has four main stages after the final agreement of the contract: mechanical designing, robot programming, factory acceptance testing (FAT), and commissioning of the robot at the customer’s site. In the factory acceptance tests, the Smelt Spout Robot is tested at the testing environment that mimics the smelt spout area of the customer’s recovery boiler. There, the functionality of robot programs is tested and modified to a specific environment using a real robot. Due to the differences in customer environments, the robot’s movements need to be always adjusted. The commissioning of the Smelt Spout Robot cell in the test environment takes approximately four days. One of these days goes teaching the points of the movements. Although the robot program is adjusted in the test environment, the movements will also need to be modified during the re-commissioning on the customer’s site. The re-commissioning takes approximately one day.

This research is a part of a huge development process where the future target is that the commissioning of the Smelt Spout Robot can be done on site without high understanding of robotics. The research is made for the Recovery Boiler (RB) Smart Business Unit that belongs to the ANDRITZ Oy Recover & Power (KRP) division. ANDRITZ is a global technology company and one of the international market leaders in various industries such as the pulp and paper industry (Andritz 2021). ANDRITZ PULP & PAPER is an internationally leading supplier of complete mills and systems for all types of pulp and paper production.

1.2 Research problem

The order backlog of the Smelt Spout Robots is growing all the time, but the commissioning is still time consuming and difficult. One of the reasons is the robot program, which is produced by traditional methods. The modifications in the programs require a thorough understanding about the structure of the program because the relationships between the parameter changes have not been considered. In practice, it means that a robot expert must always be involved in the commissioning process, which increases the costs of the project. The situation also makes it difficult to schedule projects around the world if the robot experts are not available. Therefore, the commissioning must be speeded up and simplified for the future.

These circumstances and practical problems raised the question of how the commissioning of the Smelt Spout Robot can be optimized. At the same time, due to the Industry 4.0, utilization of the virtual tools is growing rapidly. There are available tools such robotic simulations that are offering new opportunities and ways to develop the robot programs as was expressed earlier. These tools are not yet part of Smart Business Unit’s robotics development projects. Therefore, this research is focusing on the research problem of how the robotic simulation can be utilized to optimize the commissioning of the Smelt Spout Robot.

1.3 Objectives and research questions

The research is worthwhile for many reasons. First, the aim is to halve the time it takes to teach the points in the test environment of the projects. Second, by identifying the possibilities of the robotic simulation, the aim is to parameterize the robot programs which decreases the need for the robotic expert on site in the future. By parameterization, the level of competence in robotics required on site can be reduced from level 1 to level 2 (Chapter 1.1). Third, although the research is focusing on the Smelt Spout Robot, it can be also utilized in future robotics solutions. The research problem is approached by looking for answers to the following research questions:

- How the simulation can be utilized to simplify and parameterize robot programs?

- Which real-world factors should be considered in the simulation when programming a real robot?

- Which are the advantages and limitations of the robotic simulation and how they affect the accuracy?

The success of the research can be measured by analyzing the results of the case where the simulation program is transferred to the real robot. By comparing the case results with the present can be ascertained how many points should be taught. The accuracy of the simulation can be measured by moving the tool of the robot to a specific location and measuring the distance to the selected object in the simulation environment and in a real life. By testing the functionality of the parameterization, the decrease of the level of competence can be evaluated.

1.4 Research methods and content limitations

The research will be carried out in three parts: literature review, robotic simulation, and a case that includes practical tests. The literature review includes the theoretical background of robot programming and its mathematical background. It is also introducing a relevant information about robotics, and different robotic simulation applications. The literature review is based on the peer-reviewed scientific publications and books. The background of the robot programming is introduced focusing on KUKA Robot Language, therefore it is based on the manufacturer’s manual: KUKA System Software (KSS) Operating and Programming Instructions for System Integrators. In the second part, the opportunities for the robotic simulation are introduced by focusing on KUKA Sim 4.0 simulation software and its features.

The final part includes the case, where the working area of the Smelt Spout Robot is created in the simulation environment based on the instructions that will be introduced in the second part. In the simulation environment, the current programs of the robot will be parametrized by utilizing the literature review and programming manual. Finally, the parametrized programs will be tested in the test environment with a real robot which model will be KUKA KR 250 R2700-2 C. The literature review of the research and the opportunities for the robotic simulation are expressed in the Chapter 2. The case is examined in more detail in the Chapter 3, which also presents new developments steps and the results of the research.

It is important to note that the robot commissioning includes a lot of work which requires a high level of knowledge of robotics, such as a PLC (Programmable Logic Controller) programming. Therefore, this research is focusing only on expediting and simplifying the part of the commissioning where the points of the movements are taught. In addition, this research is focusing on only one robotic simulation software, which is why it cannot be fully completely reproduced with other simulation software.

2. Review of robotic simulation

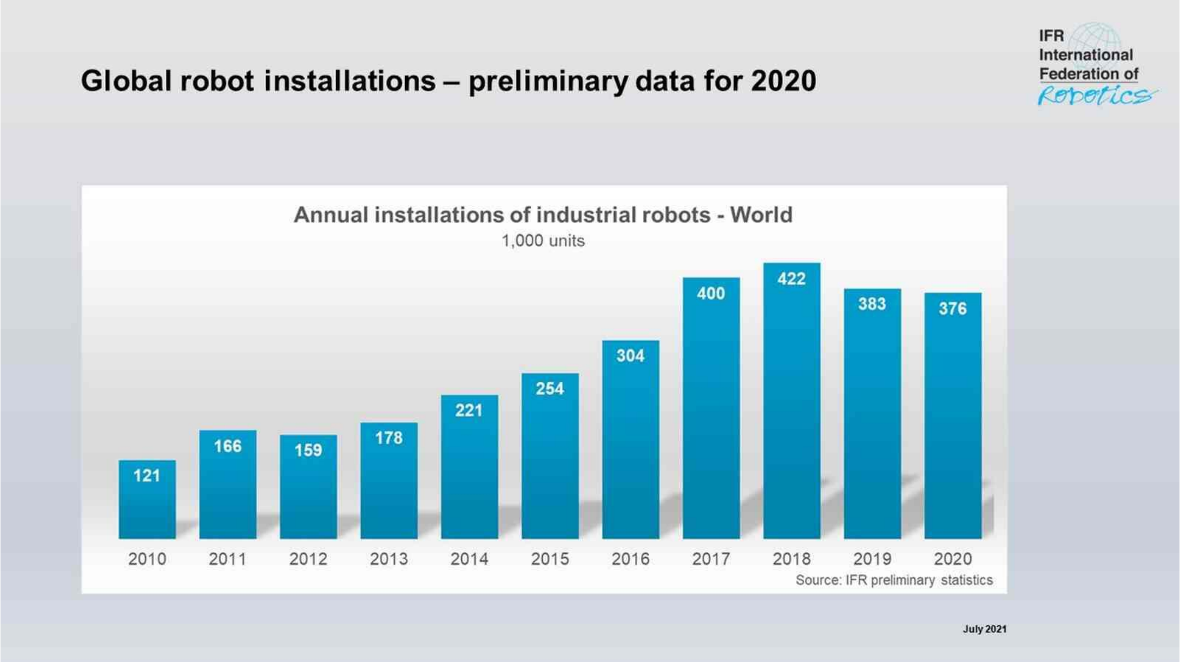
As was expressed in the Chapter 1.4, the research methods are divided into three parts: literature review, robotic simulation, and the case. This chapter introduces the literature review part, and the features of the robotic simulation focusing on KUKA Sim 4.0 simulation software. The literature review introduces the history of the robotic simulations. It also offers a theoretical background of the motion programming. These parts are based on the peer-reviewed scientific publications and books. Finally, it presents the basics of motion programming in KUKA Robot Language by based on the KUKA's programming manual. The Chapter 2.2 describes the features of KUKA SIM 4.0 by providing instructions for creating a simulation environment for the Smelt Spout Robot. The instructions are based on the practical experience of the author of this research and the software manual.

2.1 Theoretical background

The industrial robots can be determined as multi-purpose manipulators that can be controlled automatically (ISO 8373 2012). They can be categorized as Linear robots, SCARA robots, Articulated robots, Parallel/Delta robots and Cylindrical robots according to their mechanical structure (IFR 2019). The movements of the robots can be modified by reprogramming. Usually, they are programmable in three or more axes (ISO 8373 2012.)

The first industrial robot was developed in 1959 by George Devol and Joseph Engelberger (IFR 2021b). In 1973, KUKA developed the first industrial robot with six electromechanically driven axes (KUKA 2021a). Although the industrial robots have a long history, they have not become more common until this millennium. According to International Federation of Robotics (IFR 2019), the annual installations of industrial robots have increased significantly in last decade, as is shown in the Figure 2. Currently, there are many robot manufacturers such as ABB, the Yaskawa Electric Corporation, Midea Group (KUKA), and the Fanuc Corporation. Although the number of installations decreased by 2

% in 2020 due to the COVID-19 pandemic, the outlook for the future looks promising. (IFR 2021a.)



Global robot installations – preliminary data for 2020

IFR International Federation of Robotics

Source: IFR preliminary statistics

July 2021

Figure 2. Global robot installations between the years 2010 – 2020 (IFR 2021a).

Robot programming can be divided into online and offline programming. In the online programming, a motion program is created by manipulating a real robot to specific points using a teach-pendant and then storing the parameters of the points for the movements. (Holubek et al. 2014, p. 63.) This conventional programming method requires an understanding about the limits and directions of the axes. The method is time-consuming because each movement must be tested several times and the accuracy of the movements depends on the skill level of the robot programmer. (Pan et al. 2012, p. 93.) One huge disadvantage of this programming method is that the robot cannot be used for the production until the program has been tested and debugged (Chang et al. 2021, p. 34).

The importance of offline programming (OLP) has grown with the increasing use of robots. This method includes alphanumeric, graphic, and know-based robot programming, which means that, for example, 3D models and 2D drawings of the layout are used to generate the program for the robot. Existing programs can be adapted to the new environment by changing the motion parameters based on the measurements. (Pan et al. 2012. p. 90.) The offline programming also includes simulation tools where the robot program can be created, executed, and tested without access to the real robot. This facilitates commissioning of the robot, as the program can be debugged and adjusted in advance. (Holubek et al. 2014, p. 63.)

The word "simulation" includes many approaches, such as virtual reality (VR), augmented reality (AR) and digital twins (DT). The number of publications of the different simulation approaches has grown rapidly in recent decades, as is expressed in the Figure 3. (de Paula Ferreira 2020, p. 5.) The first robotic simulation applications were developed more than 20 years ago. One of them is ABB's RobotStudio. (ABB 2002, p. 552.) These first versions were mainly used to visualize the process to customers before the physical installation due to the low accuracy of the movements. (Perez et al. 2020, p. 3633.) Currently, there are numerous robotic simulation software available. Some of them are expressed in the Table 1. Each one of the major robot manufacturers has its own simulation software. There are also available several options offered by other suppliers.

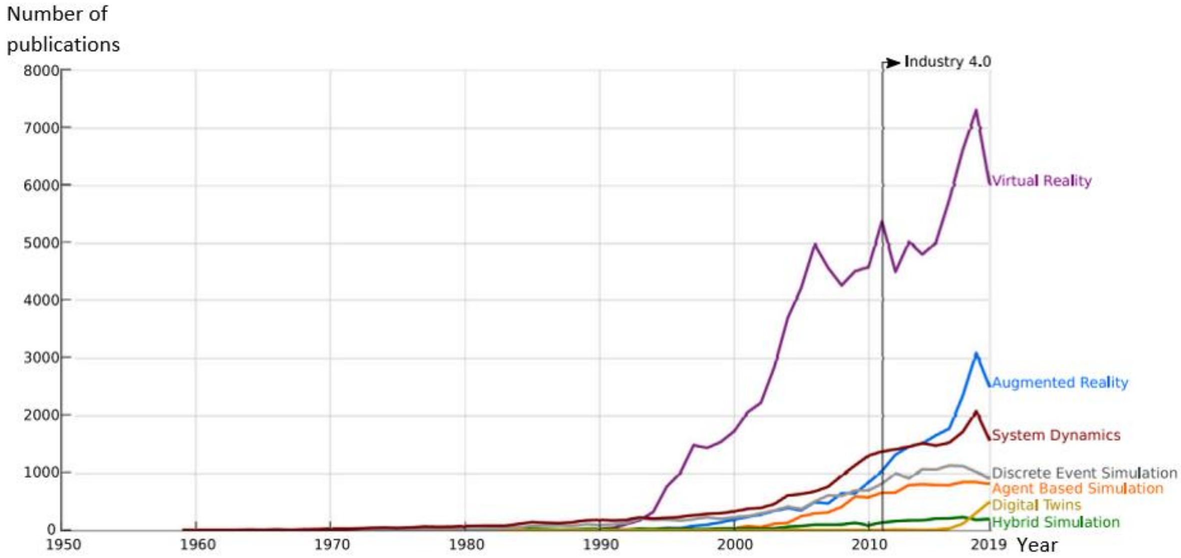


Figure 3. Number of publications of simulation approaches (de Paula Ferreira 2020, p. 6).

Table 1. List of commercial robotic simulation software.

Simulation software	Description
ABB RobotStudio	Most used offline programming software (ABB 2021).
Fanuc ROBOGUIDE	Popular simulation software for Fanuc robots (Fanuc 2021).
Delfoi robotics software	Simulation software for major robot brands (Delfoi 2021).
KUKA Sim 4.0	Introduced in the Chapter 2.2.
MotoSim EG-VRC	Offline programming software for Yaskawa robots (Yaskawa 2021).
RoboDK	Supports for many robot controllers (RoboDK 2021).
Visual Components	Three versions available with different features (Visual Components 2021).
Work Space	Capability for 3D CAD modelling (W.A.T. Solutions 2021).

The utilization of robotic simulations has increased due to its numerous advantages (de Paula Ferreira 2020, p. 13). Their main purpose is to visualize the performance of the robot by simulating the theoretical model of it (Datteri & Schiaffonati 2019, p. 117). However, these days the robotic simulations are not only used for the visualizing but also for the robot programming, creating a digital twin of the production cells and therefore optimizing the real installation and commissioning in advance in the virtual world (Ružarovský 2019, p. 1). They can be used to analyze and test complex systems during the development of a new product (de Paula Ferreira 2020, p. 14). By visualizing the robot programs for the other engineers who are unfamiliar with the robot movements, the robot cell can be optimized already at the design stage when the required space by the robot is known (Scheidegger et al. 2018, p. 475).

Simulations tools not only provide a way to test the robot movements, they can be also used to learn them. The simulation is quite effective teaching tool in robotics because there is not always possible to purchase a real robot and test its performance. In addition, it requires time to learn how to control a real robot. Therefore, a risk-free environment where damage cannot occur is suitable for teaching. (González-García 2019, p. 926.) The simulation tools are not only for the robot movements simulation, but they also enable to simulate the performance of the whole process in advance to expedite the actual commissioning, which is better known as virtual commissioning (VC). There, the functionality of the PLC programs and I/O (Input/Output) control behavior of the security circuit can be tested (Ružarovský 2019, p. 6).

By combining VR technology and robot simulations also the human-robot collaboration can be inspected (Perez et al. 2019, p. 114-119).

Although the simulation has significant advantages, there are also some challenges. The simulation software includes numerous features that takes time and courses to learn (González-García 2019, p. 929). In addition to understanding basics of robotics and coordinate systems, simulation also requires an understanding about programming, 3D modeling, and automation. It brings together many disciplines, so there are not many professionals available. Therefore, the salaries of the simulation engineers are also high (de Paula Ferreira 2020, p. 6) which affects companies' interest in simulations.

Mathematical background of robot programming

As was expressed earlier, robot movements are based on the parameters of the stored points. To understand the significance of these parameters, their mathematical background must be examined. Theoretically, the industrial robot can be defined as a multi-body system (MBS) (Flores 2015, p. 1) consisting of bodies connected to each other by kinematics joints. In robotics, the purpose is to move the endpoint of the robot to the specific location by actuating these bodies by electric and hydraulic motors (Bajd et al. 2010, p. 1). In more detail, the bodies can be examined as particles that have specific distances to the reference point. Due to the distance between these particles is not changing, they also have the same speed during the motion. Therefore, the industrial robot can be also specified as a rigid body multi-body system. (Shabana 2005, p. 1-3.)

The positions of the particles are usually expressed using the Cartesian coordinates X, Y and Z with respect to frame of reference. Therefore, it can be said that the particle has three degrees of freedom (DOF) (Bajd et al. 2010, p. 2). In the Figure 4, the P describes the particle, and the vector **u** is the three-dimensional position vector that can be expressed as (Shabana 2005, p. 3):

$$\mathbf{u} = [u_X \quad u_Y \quad u_Z]^T \tag{2.1}$$

where, u_X, u_Y and u_Z are the components of the vector **u**.

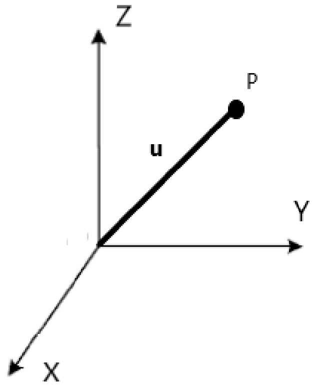


Figure 4. The location of the particle P can be expressed with respect to the reference frame by the vector **u**.

The rigid bodies consist of at least three particles (Bajd et al. 2010, p. 3). Usually, when expressing the configuration of the rigid bodies, two kind of coordinate systems are needed. The first is a fixed coordinate system that represents the origin of the entire system as a function of time. The second, also known as a body frame of reference, describes the location of the body with respect to the fixed frame. The relationship of these coordinate systems is expressed in the Figure 5, where the vector **R** is a three-dimensional vector that can be written as the vector **u** (Shabana 2005, p. 3):

$$\mathbf{R} = [R_X \quad R_Y \quad R_Z]^T \tag{2.2}$$

Due to the rigid body can be rotated with respect to the global frame, three additional coordinates ϕ, θ and ψ are required to describe the orientation of it (Flores 2015, p. 16). Thus, the configuration of the rigid body can be expressed in three translational and three rotational coordinates. Therefore, the rigid body has six degrees of freedom. (Bajd et al. 2010, p. 3.)

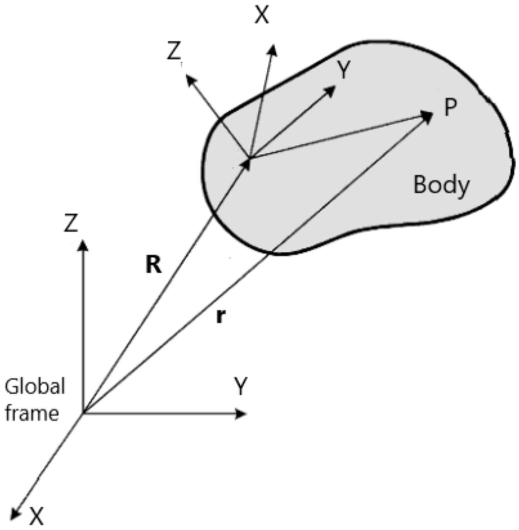


Figure 5. The location of the particle P can be expressed with respect to the global frame with a help of the body frame of reference.

Thus, the position vector of the particle *P* can be expressed with respect to the global frame as:

$$\mathbf{r} = \mathbf{R} + \mathbf{A}\mathbf{u} \tag{2.3}$$

where, **A** is a 3 x 3 rotation matrix that can be formed, for example, using Euler Angles. It is the method to describe the orientation of a rigid body. (Flores 2015, p. 15.) According to the Euler’s theorem, there are 12 different ways to describe rotations: X-Y-Z, X-Z-Y, X-Y-X, X-Z-X, Y-Z-X, Y-X-Z, Y-Z-Y, Y-X-Y, Z-X-Y, Z-Y-X, Z-X-Z and Z-Y-Z (Bajd et al. 2013, p. 26). Each rotation can be expressed mathematically with the rotation matrix. The rotation about Z axis can be written as (Flores 2015, p. 16):

$$\mathbf{A_1} = \begin{bmatrix} \cos(\phi) & -\sin(\phi) & 0 \\ \sin(\phi) & \cos(\phi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{2.4}$$

The rotation about Y axis can be expressed as:

$$\mathbf{A_2} = \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \tag{2.5}$$

And the rotation about X axis can be expressed as:

$$\mathbf{A_3} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\psi) & \sin(\psi) \\ 0 & \sin(\psi) & \cos(\psi) \end{bmatrix} \tag{2.6}$$

It is important to note that the final orientation of the rigid body depends on the order of rotations (Flores 2015, p. 17). The rotations are visualized in the Figure 6 using a numerical example. First, the coordinate system XYZ is rotated 90 degrees about Z axis. Then the coordinate system X’Y’Z’ is rotated -45 degrees about once rotated axis Y’. Finally, the coordinate system X’’Y’’Z’’ is rotated 30 degrees about twice rotated axis X’’. The orientation of three successive rotated coordinate system is expressed as X’’’Y’’’Z’’’ on the right in the Figure 6.

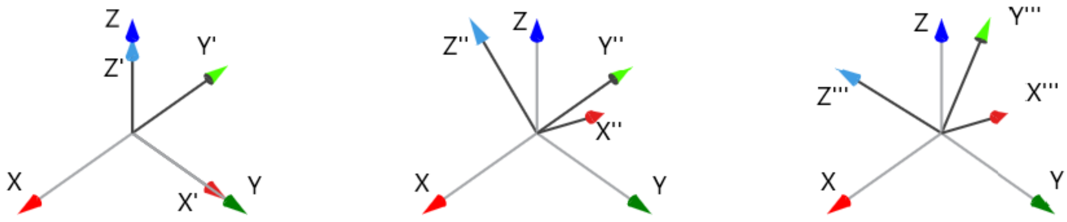


Figure 6. Three successive rotations of coordinate system using ZYX Euler convention (Bonev 2017).

The rotational transformation matrix for the three successive rotations can be determined as follows (Bajd et al. 2013, p. 22.):

$$\mathbf{A} = \mathbf{A_1A_2A_3} = \begin{bmatrix} \cos(\phi) & -\sin(\phi) & 0 \\ \sin(\phi) & \cos(\phi) & 0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos(\theta) & 0 & \sin(\theta) \\ 0 & 1 & 0 \\ -\sin(\theta) & 0 & \cos(\theta) \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos(\psi) & \sin(\psi) \\ 0 & \sin(\psi) & \cos(\psi) \end{bmatrix}$$

$$\mathbf{A} = \begin{bmatrix} \mathrm{c}(\phi)\,\mathrm{c}(\theta) & \mathrm{c}(\phi)\,\mathrm{s}(\theta)\,\mathrm{s}(\psi) - \mathrm{s}(\phi)\,\mathrm{c}(\psi) & \mathrm{c}(\phi)\,\mathrm{s}(\theta)\,\mathrm{c}(\psi) + \mathrm{s}(\phi)\,\mathrm{s}(\psi) \\ \mathrm{s}(\phi)\,\mathrm{c}(\theta) & \mathrm{s}(\phi)\,\mathrm{s}(\theta)\,\mathrm{s}(\psi) + \mathrm{c}(\phi)\,\mathrm{c}(\psi) & \mathrm{s}(\phi)\,\mathrm{s}(\theta)\,\mathrm{c}(\psi) - \mathrm{c}(\phi)\,\mathrm{s}(\psi) \\ -\mathrm{s}(\theta) & \mathrm{c}(\theta)\,\mathrm{s}(\psi) & \mathrm{c}(\theta)\,\mathrm{c}(\psi) \end{bmatrix} \quad (2.7)$$

where, c describes the cosine function and s is the sine function. The transformation matrix is used to determine the orientation the points relative to the different coordinate systems. Due to the body orientation depends on the order of the rotations, also the significance of the Euler Angles is changing. Therefore, the transformation matrix is also varying depending on the Euler convention. The understanding of the Euler Angles is not simple, thus the Figure 7 is used to visualize the rotations in the same figure.

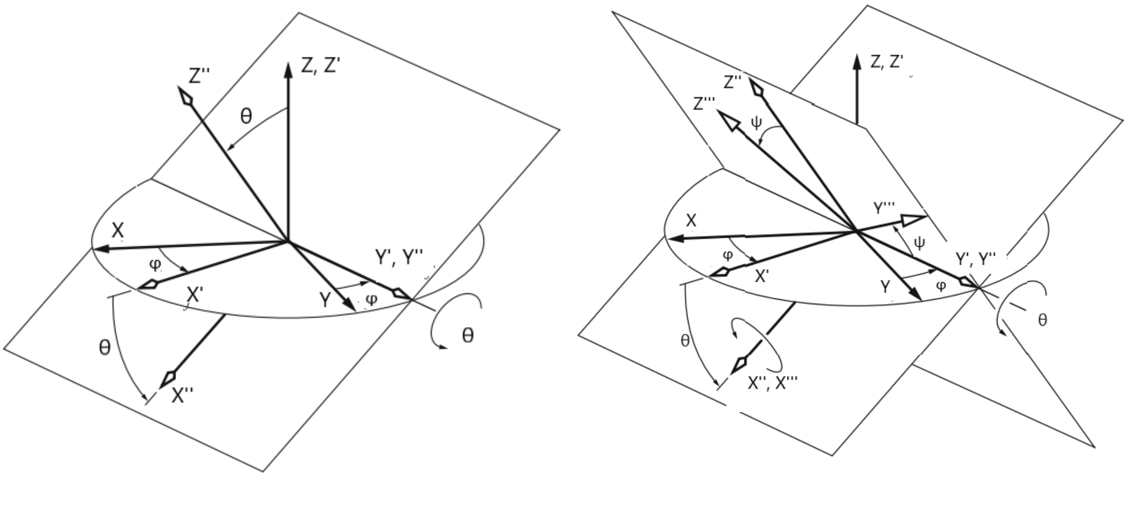


Figure 7. The Euler Angles ϕ, θ and ψ in ZYX Euler convention (Modified: Briot & Khalil 2015, p. 29).

The rotation matrix is also useful in the inverse problems where the purpose is to define the rotation of the rigid body (Bajd et al. 2013, p. 13). It can be written in general as:

$$\mathbf{A} = \begin{bmatrix} X_x & Y_x & Z_x \\ X_y & Y_y & Z_y \\ X_z & Y_z & Z_z \end{bmatrix} \tag{2.8}$$

where, X, Y and Z vectors are expressed by the components (Briot & Khalil 2015, p. 30). Therefore, it can be written that:

$$\begin{bmatrix} \mathrm{c}(\phi)\,\mathrm{c}\,(\theta) & \mathrm{c}(\phi)\,\mathrm{s}(\theta)\,\mathrm{s}(\psi) - \mathrm{s}(\phi)\,\mathrm{c}\,(\psi) & \mathrm{c}(\phi)\,\mathrm{s}(\theta)\,\mathrm{c}(\psi) + \mathrm{s}(\phi)\,\mathrm{s}(\psi) \\ \mathrm{s}(\phi)\,\mathrm{c}(\theta) & \mathrm{s}(\phi)\,\mathrm{s}(\theta)\,\mathrm{s}(\psi) + \mathrm{c}(\phi)\,\mathrm{c}(\psi) & \mathrm{s}(\phi)\,\mathrm{s}(\theta)\,\mathrm{c}(\psi) - \mathrm{c}(\phi)\,\mathrm{s}(\psi) \\ -\mathrm{s}\,(\theta) & \mathrm{c}(\theta)\,\mathrm{s}(\psi) & \mathrm{c}(\theta)\,\mathrm{c}(\psi) \end{bmatrix} = \begin{bmatrix} X_x & Y_x & Z_x \\ X_y & Y_y & Z_y \\ X_z & Y_z & Z_z \end{bmatrix} \tag{2.9}$$

According to the Figure 7 and the Equation 2.9, the Euler Angles can be determined, for example, as follows:

$$\phi = \mathrm{atan2}\left(\frac{X_y}{X_x}\right) \tag{2.10}$$

$$\sin(\theta) = -X_z \tag{2.11}$$

$$\cos(\theta) = \cos(\phi)\, X_{\overline{P_X}} + \sin(\phi) Y_{\overline{P_X}} \tag{2.12}$$

$$\theta = \arctan\left(\frac{\sin(\theta)}{\cos(\theta)}\right) \tag{2.13}$$

$$\sin(\psi) = \sin(\phi)\, Z_{\overline{P_X}} - \cos(\phi)\, Z_{\overline{P_Y}} \tag{2.14}$$

$$\cos(\psi) = -\sin(\phi)\, Y_{\overline{P_X}} + \cos(\phi)\, Y_{\overline{P_Y}} \tag{2.15}$$

$$\psi = \arctan\left(\frac{\sin(\psi)}{\cos(\psi)}\right) \tag{2.16}$$

These angles can be used to determine the orientation of the rigid body when the ZYX Euler convention is used.

Overview of motion programming in KUKA Robot Language

The robot motion programming is based on the mathematical background that was expressed in the previous chapter. The purpose of this chapter is to give a basic information of the motion programming using KUKA Robot Language (KRL). The information is based on the Operating and Programming Instructions for System Integrators of KSS 8.6. The chapter focus on programming the industrial robot that has six axes. The structure of 6 DOF industrial robot is introduced in the Figure 8. It consists of rigid bodies connected to each other by joints A1, A2, A3, A4, A5 and A6, which can be rotated separately. The limits of these joints depend on the robot model and they are different compared with each other (Table 2). These limits restrict the robot’s movements and should therefore be considered when designing a robot program.

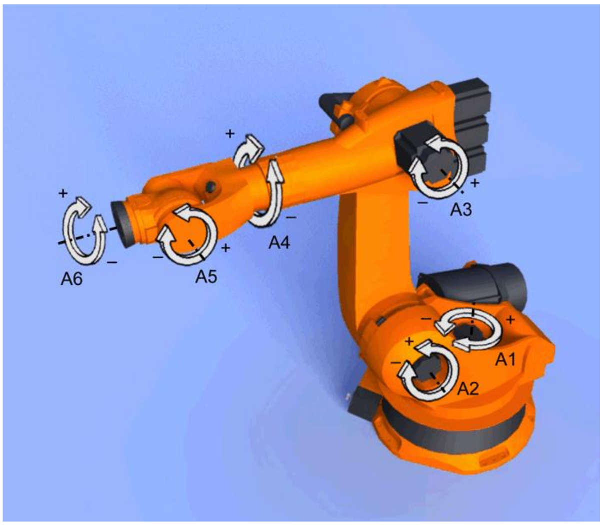


Figure 8. Joints of the 6 DOF industrial robot (KUKA 2021e, p. 85).

Table 2. The joints limit of the KUKA KR 250 R2700-2 C that is examined in this research (KUKA, 2021b).

Axis	Motion range
A1	± 185 °
A2	- 140 ° / -14 °
A3	-120 ° / 168 °
A4	± 350 °
A5	± 122.5 °
A6	± 350 °

In practice, the control of the robot based on the kinematics that allows to obtain the location of the end point of the robot's tool and the angles of joints relative to each other, as is expressed in the Figure 9 (Perez et al. 2019, p. 116).

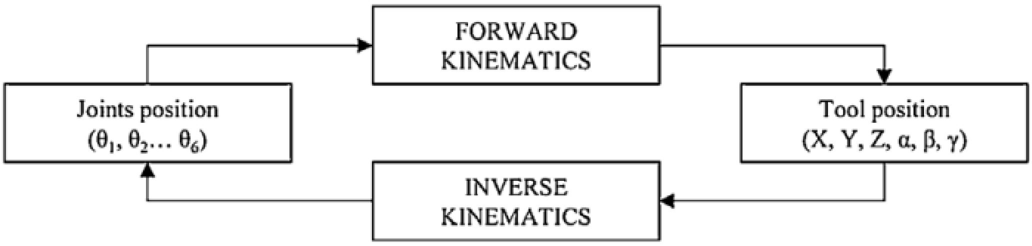


Figure 9. The control of the robot is based on kinematics (Perez et al. 2019, p. 116).

In KRL, the location of a tool coordinate system, $TOOL, is expressed with respect to the reference frame. Usually, the tool coordinate system is determined to the endpoint of the tool also known as Tool Center Point (TCP). The tool coordinate system describes the position of the end effector of the tool. It is an offset from the center point of the mounting flange of the robot. It can be defined numerically utilizing parameters or using calibration methods such as XYZ 4-point method where the TCP is moved a specific point from four different directions. (KUKA 2021e, p. 83 - 198.)

The location of the TCP can be expressed with respect to the global frame, $WORLD, or the base coordinate system, $BASE. The global frame, also known as a world coordinate system, is a fixed frame that describes the origin of the entire system. The location of the robot with respect to the world coordinate system is expressed as $ROBROOT. By default, the world and root coordinate systems of the robot are defined at the same point. (KUKA 2021e, p. 83). In several robot cases, the world coordinate systems of the robots are often defined to the same point to facilitate the communication between the robots.

The base coordinate systems are used to determine the locations of the workpieces as is shown in the Figure 10. They are simplifying the programming because the motion points can be defined as a transition from the workpiece. (KUKA 2021e, p. 189.) As the workpiece moves, the points can be re-taught simply by moving the base coordinate system with it. Therefore, most of the points are usually defined with respect to the base coordinate systems because it facilitates the transformation of the movements due to the changes in the environment. The base coordinate systems can be determined numerically as an offset from the world coordinate system or manipulating the robot's tool to the specific points and using different calibration methods. One of them is 3-point method, where the tool is moved to the origin and 2 specific points. (KUKA 2021e, p. 189-204.) Different coordinate systems of the KRL are summarized in the Figure 10.

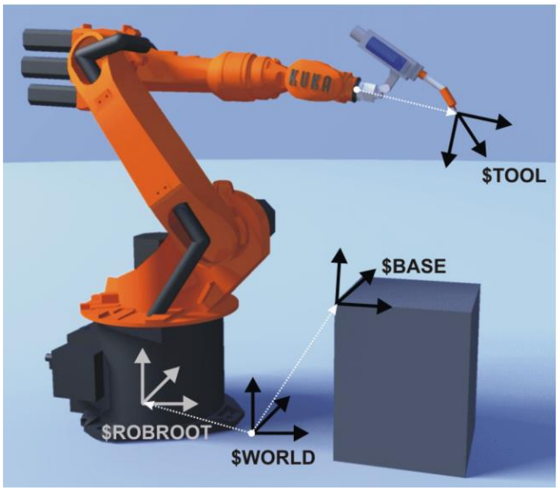


Figure 10. Different coordinate systems in KUKA Robot Language (KUKA 2021e, p. 84).

As was discussed in the previous chapter, the configuration of the rigid body can be expressed by three translational and three rotational coordinates. In KRL, these coordinates are expressed as X, Y, Z, A, B and C. The translational coordinates X, Y, Z determine the location of the TCP with respect to the cartesian coordinate system, and the rotational coordinates A, B and C express the orientation of the TCP with respect to the same frame of reference. According to the KUKA System Software manual, KUKA describes these three rotations as is expressed in the Figure 11. (KUKA 2021e, p. 211.) In the KRL, the rotations should be determined in order A, B and C, therefore KUKA uses the ZYX Euler convention.

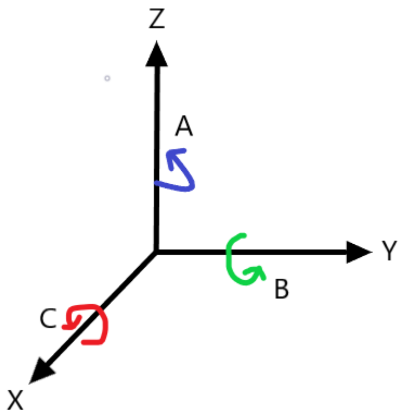


Figure 11. In KUKA Robot Language, the Euler Angles are expressed as A, B and C.

The location of the TCP can be expressed using predefined data types E6POS and POS which are expressed in the Table 3 (KUKA 2021e, p. 540).

Table 3. Data types for the motion programming in KRL (KUKA 2021e, p. 540).

Data type	**Variables of the datatype**
E6POS	{X, Y, Z, A, B, C, S, T, E1, E2, E3, E4, E5, E6}
POS	{X, Y, Z, A, B, C, S, T}

As can be seen from the Table 3, the data types contain also additional variables. Due to the coordinates do not determine the angles of the axes, the additional variables Status (S) and Turn (T) are used. The Status is defining the axis angles according to the position of TCP.

The Turn determines whether the axis angles are greater than or less than zero. The Status and Turn are only used in the Point-To-Point movements that will be expressed later. The positions of the possible external axes can be determined using variables E1 – E6. Other useful data types for the motion programming are AXIS, E6AXIS and FRAME (Table 4), where A1 – A6 describe the axes positions (KUKA 2021e, p. 540.):

Table 4. Useful data types for motion programming in KRL (KUKA 2021e, p. 540).

Data type	**Variables of the datatype**
AXIS	A1, A2, A3, A4, A5, A6
E6AXIS	A1, A2, A3, A4, A5, A6, E1, E2, E3, E4, E5, E6
FRAME	X, Y, Z, A, B, C

The AXIS and E6AXIS can be used to determine the orientation of the robot axes. For example, the home position of the robot is defined as E6AXIS data type. FRAME data types can be used to define additional coordinate systems to help the robot programming. The movements of the robot are determined in a module that consists of a SRC file and a DAT file. The SRC file includes the different types of motion commands and the actual code of the program. The DAT file includes the data for the movements such as the coordinates of the points. (KUKA 2021e, p. 538-540.)

The KRL motion types can be divided into point-to-point motion (PTP), linear motion (LIN), circular motion (CIRC) and spline motion. The PTP motion is used to execute the fastest path. It is not widely used in the area where are many obstacles because the movement between two points cannot be predicted and thus the collisions cannot be avoided. The LIN motion commands can be used to define the movement on a straight line between two points which makes it easier to avoid collisions. The motion type CIRC is used to create circular motion between the start and end point. Spline motion is an advantageous motion type which can be used to replace classic motions as SPTP, SLIN and SCIRC. (KUKA 2021e, p. 407-413.)

The spline motion type has significant advantages compared with the classic motion type. First, the physical limits of the robot are considered during the motion planning. It means that when using the classic motions, the robot may stop suddenly during the motion execution due to the physical limits. In the spline motions, the physical limits of the robot may reduce the programmed velocity, but the number of unexpected stops can be minimized. Second, an exact positioning can be used easily with the spline motions, therefore the path is always the same. Classic motions are often defined using an approximate positioning because it is complicated and time-consuming to determine the exact path due to the physical limits are ignored. The approximate positioning means that the TCP does not reach exactly the programmed point as in the exact positioning, as is shown in the Figure 12. Third, there could be some unexpected reduction of the velocity in the classic motions. (KUKA 2021e, p. 414-415.)

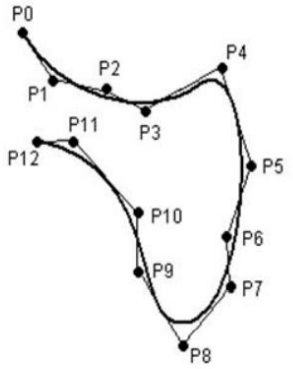


Figure 12. The movement path can be defined using an approximate positioning or an exact positioning (KUKA 2021e, p. 414).

The motion of the robot can be also determined using the relative motion programming which is based on defining the end point of the motion relative to the current position. The relative motion commands can be used with the classic and spline motions as: PTP_REL, LIN_REL, CIRCL_REL, SPTP_REL, SLIN_REL, SCIRCL_REL and SPL_REL. The relative motions can be executed using the directions of the base coordinate system or the tool coordinate system, as shown in the Table 5. Relative movements using the tool coordinate system can be also executed by utilizing the $POS_ACT system variable that is determining the actual position of the active tool in the active frame of reference (KUKA 2021e, p. 550-641.)

Table 5. Relative motion command can be executed using the directions of the active base coordinate system or the tool coordinate system or using $POS_ACT system variable (KUKA 2021e, p. 550-641).

Motion command	**Used coordinate system**
SLIN_REL {X -150, Z 200, A 3, C 45}	Base
SLIN_REL {X -150, Z 200, A 3, C 45} #TOOL	Tool
SLIN $POS_ACT : { X -150, Y 0, Z 200, A 3, B 0, C 45}	Tool

As can be noticed from the Table 5, the relative motion command can be executed by written only the transformation parameters. When determining the motion command relatively to the current position using $POS_ACT system variable, all the parameters should be written, even if the value would be zero. In addition, a geometric operator that is symbolized by a colon ":" should be used with the actual position variable, as was shown in the Table 5. (KUKA 2021e, p. 641-642.)

The geometric operator is a useful tool to add calculated positions which are determined geometrically. It is also known as a "frame operation" because it can be used to link data types such FRAME and POS in the motion commands. (KUKA 2021e, p. 641.) The mathematical background of the geometric operator is expressed in the Table 6 where the vector **r** is visualized in the Figure 13.

Table 6. A geometric operator is a tool to link the vectors.

Form of representation	**Vector**
Mathematical	**r = R + Au**
KUKA Robot Language	**r = R : u**

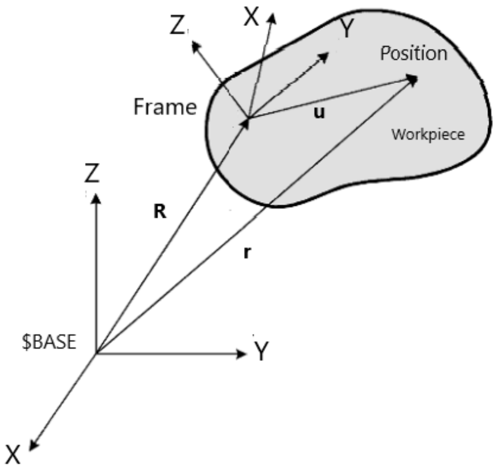


Figure 13. A geometric operation can be used to link data types such as frames and positions.

As was expressed earlier that the location of the TCP can only be determined directly with respect to the world or base coordinate systems. Due to the benefits of the geometric operator, the points can be also defined with respect to the frames as was shown in the Figure 13.

2.2 Overview of KUKA Sim 4.0

According to KUKA, the KUKA Sim 4.0, that was released to a market in the summer 2021, is an intelligent simulation software that was developed in a future-oriented way. It is a virtual tool for optimizing robot commissioning by creating a digital twin of the process. (KUKA 2021c.) In this research, the KUKA Sim 4.0.1 is examined. The user interface is presented in the Appendix I, which shows the different tabs (Home, Modelling, Program & Drawing) of the software.

Usually, the robots are working in a specific environment that determines the size of the robot and its movements. Therefore, the logical way to start a simulation is to create a simulation environment that can be done in several ways. First, eCatalog – a component library includes the basic 3D shapes such as blocks, boxes, planes, and pallets which dimensions can be modified. Thus, it can be used to create simple environments. Second, the efficient way to mimics the real environment is to model it using 3D CAD software and then import it into the simulation software. There are available numerous supported CAD files such as a STEP file and SolidWorks File (SLDPRT). Third, the real environment can also be imported as a point cloud (Figure 14) that is effective way to show the environment with small details.

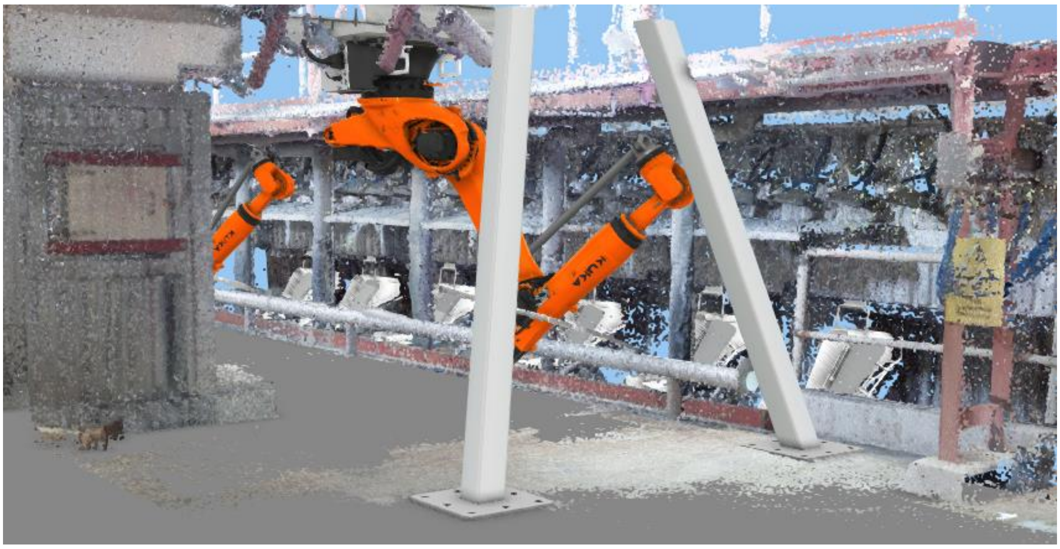


Figure 14. Utilizing 3D CAD models and a point cloud, it is possible to create a simulation environment that includes the smallest details of the real environment.

In the virtual world, objects can be placed relative to each other very accurately. The design of the environment is facilitated by the origin of the virtual world, which is the frame of reference for the layout. The body frame of reference can be determined for each component which facilitates to determine its location. The body frame of reference can be determined using Move and Snap Origin features that can be found on the Home tab (Appendix I). The entire environment can be also imported to the simulation as a whole and its origin can be determined in the same way. The location of the body frame of reference with respect to the virtual frame of reference can be seen in Component Properties. The robot can be dragged to the simulation environment also from the eCatalog that includes almost 300 robot models. Its location can be defined to the target location using Snap and Align tools or determining the parameters with respect to the virtual world origin.

The eCatalog also includes tools for the robot such as vacuum grippers and welding guns, but the tool can be also imported as a 3D model. The tools which are added from the eCatalog can be snapped directly to A6 joint of the robot using Plug and Play (PNP) (Appendix 1: Home) feature because they have a configured connect interface. The interface must be created to user’s own tools to create a physical connection between the robot and the tool. It can be created by defining the origin of the tool to the target face and using End Effector wizard (Appendix I: Modeling). Finally, the tool can be then determined to follow the movement of the A6 joint using Attach feature. The physical connection can be created with the PNP feature as is expressed in the Figure 15.

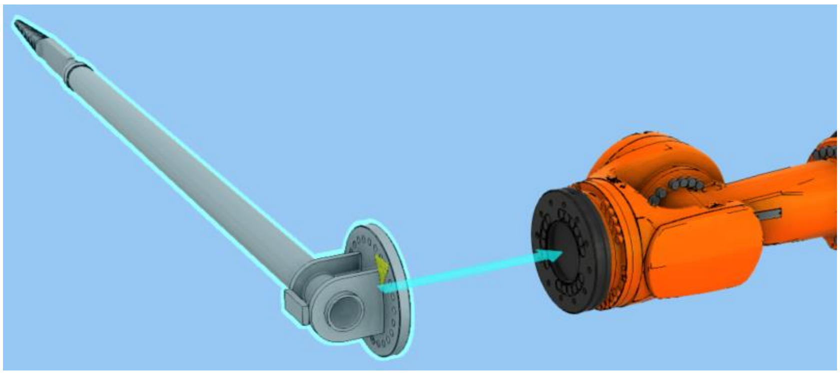
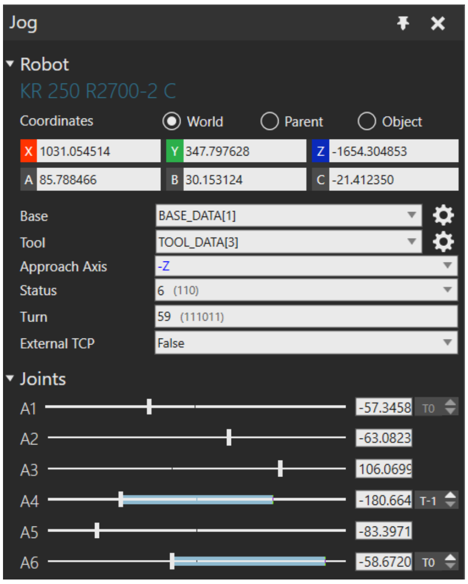


Figure 15. The blue arrow facilitates to determine the physical connection between the tool and the robot when using Plug and Play feature.

The robot programming in KUKA Sim 4.0 is based on the KRL that was introduced previously. The robot can be manipulated in Program tab (Appendix I) when the Jog tab is selected. The angles of joints can be seen and changed in Jog panel (Figure 16) where the coordinates of the TCP can be seen relative to the virtual world coordinate system (World), relative to the selected coordinate system (Parent) or relative to tool coordinate system itself (Object). It is important to understand the difference between the virtual world coordinate system and the robot’s own world coordinate system. The virtual world coordinate system is not connected to the robot controller or the motion programming such as the robot’s own world coordinate system.

The active base (BASE_DATA) and tool coordinate systems (TOOL_DATA) for the movements can be chosen from Jog panel (Figure 16). In the virtual world, these coordinate systems can be defined in numerous ways such as snapping the coordinate system to the desired location or determining the parameters with respect to the frame of references. As was expressed earlier, the frame of reference of the tool coordinate system is the centre of the robot flange and robot world coordinate system is the frame of reference for the base coordinate systems. Once the desired coordinate systems are selected, the robot manipulation can be started. It can be jogged by dragging the coordinate system that is shown in TCP (Appendix I: Program) relative to the different coordinate systems. This facilitates the test the reachability of the robot in different situations. The robot can be also manipulated by changing the joint values or coordinates in Jog panel (Figure 16).



KR 250 R2700-2 C

Coordinates ● World ○ Parent ○ Object

X 1031.054514
Y 347.797628
Z -1654.304853
A 85.788466
B 30.153124
C -21.412350

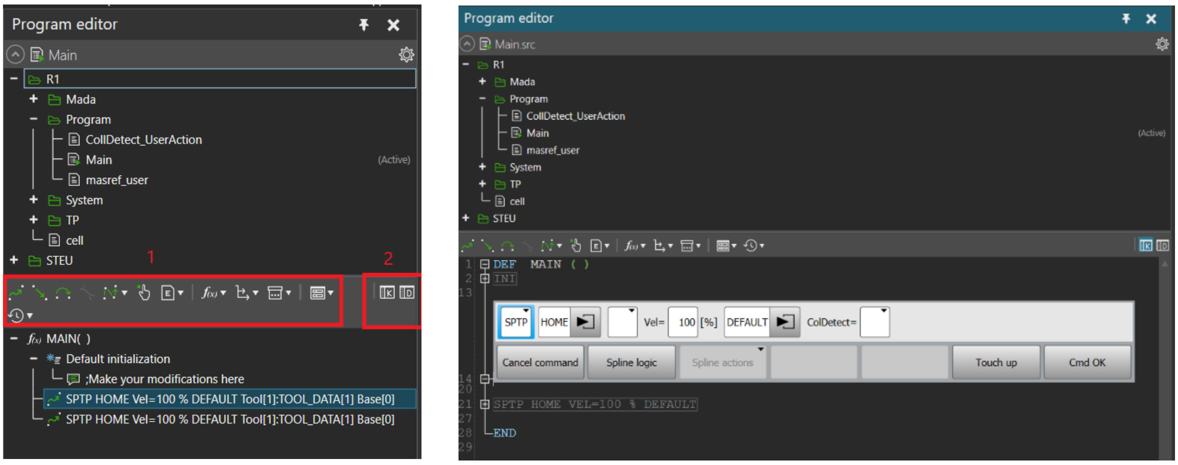
Base: BASE_DATA[1]
Tool: TOOL_DATA[3]
Approach Axis: -Z
Status: 6 (110)
Turn: 59 (111011)
External TCP: False

A1 -57.3458 T0
A2 -63.0823
A3 106.0699
A4 -180.664 T-1
A5 -83.3971
A6 -58.6720 T0

Figure 16. The current values of the robot’s joints can be seen in Jog panel.

The robot programs can be created in the Program Editor (Figure 17) where are several ways to determine the points for the movements. The simplest option for the beginners is to use a visual program tree of the KRL editor which does not require much knowledge of robot programming. The motion statements can be added by manipulating the robot, and then selecting the right statement from the Statements Toolbar (Figure 17 a: 1) that contains the basic motion programming commands, and loops such as WHILE, IF and SWITCH. The Visual program tree is an efficient way to learn the basics of robot programming but is not suitable for the advanced programs.

Another option that requires programming experience is to use the advanced KRL editor (Figure 17 a: 2), where the programs can be written in KUKA Robot Language. New motion commands can be added using the statements from the Statements Toolbar, writing the motion commands, or using inline forms as is shown in the Figure 17 b. The KRL files can also be imported directly to the software which helps to simulate the existing programs. This option requires a lot of understanding about the programming language to debugging of the program that can be time-consuming.



```
SPTP HOME Vel=100 % DEFAULT Tool[1]:TOOL_DATA[1] Base[0]
SPTP HOME Vel=100 % DEFAULT Tool[1]:TOOL_DATA[1] Base[0]
```

```
DEF  MAIN ( )
INI
SPTP HOME VEL=100 % DEFAULT
END
```

a) Visual Programming tree

b) Inline forms can be used in KRL editor.

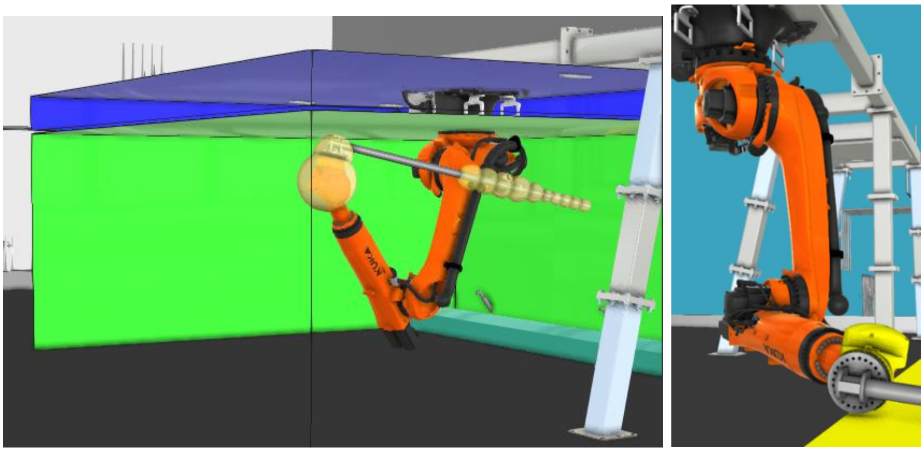
Figure 17. The Program editor in KUKA Sim 4.0.

The home position of the robot is usually the first point that should be taught because it often determines the start and the end position of the programs. It can be determined several ways such as numerically in Statement Properties or directly in the $Config.dat file. If the home position is used as the starting position in the simulation, it must be saved by pressing “Save State” from simulation settings (Appendix I: Program).

The virtual world has several advantages such as the capability to test the different robot models until the final decision. The robot can be changed by adding a new one from eCatalogue and using Exchange Robots feature (Appendix I). This allows to test the reachability and suitability of the robot for the modeled environment. In real life, replacing a robot would be expensive and could take several weeks due to the long delivery times.

Another beneficial thing in KUKA Sim 4.0 is “Move Robot World Frame” feature (Appendix I). In several robot cases where the robots are working together, their world coordinate systems are usually defined to the same point. This way the communication between the robots is easier when their locations are determined with respect to the same point. In the simulation, the robot world frame can be easily moved to the desired point by dragging the world coordinate system or specifying its coordinates with respect to the virtual frame of reference. It is necessary to note that the points which are determined with respect to the world coordinate system are also moving. This can be prevented by using “Lock Positions” feature that locks the positions of with respect to the virtual frame of reference. The positions and coordinate systems can be visualized using the visualization tools of the software.

A third feature worth mentioning is Collision Detection feature that can be utilized to seek the possible collisions to prevent damage. The collisions can be tested between the safety tools and areas which can be visualized as is shown in the Figure 18 a. The green and blue boxes are visualizing the safety monitoring spaces and the yellow spheres are illustrating the safety tool. The safety tool and safety monitoring spaces can be created from the Safety Devices (Appendix 2). The space type of the monitoring space can be axis-specific workspace, axis-specific protected space, Cartesian workspace (Figure 18 a: green) and Cartesian protected space (Figure 18, a: blue). With the axis-specific spaces, it is a possible to set the limit values for each axis and check are the values exceeded. The cartesian spaces are beneficial when the collision detection needs to be tested between the safety tool and some area. The collision detection can also be tested between the components without the safety configurations, such as between the robot and the floor. The collisions can be detected a in yellow (Figure 18 b & Appendix 3) when the Collision Detection is determined and enabled.



a) Safety areas and tools can be visualized.

b) Collisions are highlighted in yellow.

Figure 18. The collisions can be detected between the safety tool and safety area. It is also possible to detect collision between the robot and the obstacles such as a floor.

The robot programs can be executed by clicking Play button at the top of the simulation. They can be also executed by connecting the robot to KUKA's virtual controller KUKA.OfficeLite whose user interface mimics a real teach pendant (KUKA 2021d). The speed of the simulation can be controlled by a speed control, which allows errors to be found quickly. One of the best features of the robotic simulation compared with the real world is the Reset button, which immediately returns the robot to its initial position. In the traditional programming, the robot can be returned to the initial position by jogging it manually or using a precise return path.

In this research KUKA Sim 4.0 is used to optimize the commissioning of the Smelt Spout Robot. The development process will be introduced in the Chapter 3. The used version will be KUKA Sim 4.0.1 which includes software bugs related to the spline motions that will affect the development process. These bugs were fixed in the new version KUKA Sim 4.0.2, but it was released to the marker after the development process.

3. Case development and results

This chapter introduces the development process of the case where the purpose was to expedite and facilitate the commissioning of the Smelt Spout Robot. The chapter introduces the development steps, and the results from the practical tests where the functionality of the development process was assessed.

3.1 Commissioning of the Smelt Spout Robot

This chapter introduces the commissioning of the Smelt Spout Robot before the development steps. It includes the main stages of the mechanical installation and the introduction to the re-commissioning of the robot. The data were collected during the commissioning of the robot in Austria in October 2021.

The function of the recovery boiler is based on burning concentrated black liquor drops which are sprayed to the recovery boiler as is shown in the Figure 19. In the combustion reaction, the inorganic part of the black liquor melts in a char bed and flows out of the recovery boiler through the smelt spouts (Figure 20, b). The smelt is mainly consisting of sodium sulfide (Na_2S) and sodium carbonate (Na_2CO_3). (Vakkilainen 2017, p. 238-258.)

The main purpose of the smelt spout is to guide the 750 – 850 °C smelt flow from the char bed to a dissolving tank (Vakkilainen 2017, p. 238-258). The structure of the smelt spout is expressed in the Figure 20, a. It consists of the smelt spout itself, and upper and lower part of the mini hood. The smelt spouts need to be cleaned regularly because the smelt starts to solidify when the temperature is decreasing under 760 °C (Tran et al. 2006, p. 182-187). Due to the danger of the smelt spout area, the Smelt Spout Robot was developed to clean-up the spouts and the hoods regularly. The innovation has increased operator safety because previously they had to be cleaned manually by human.

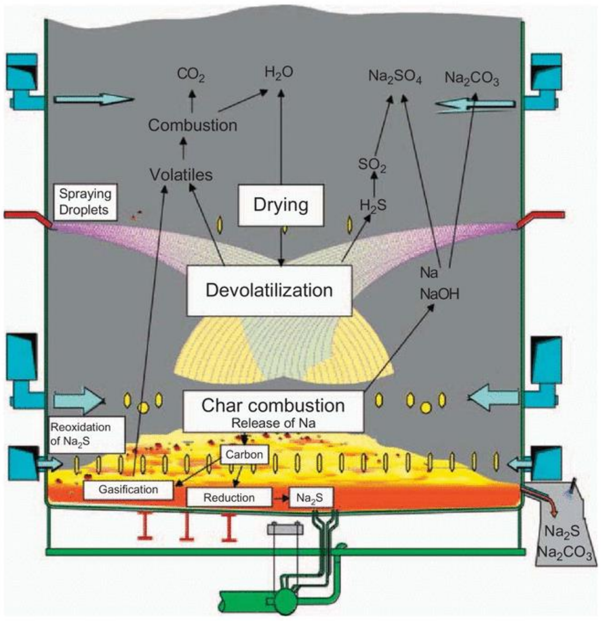
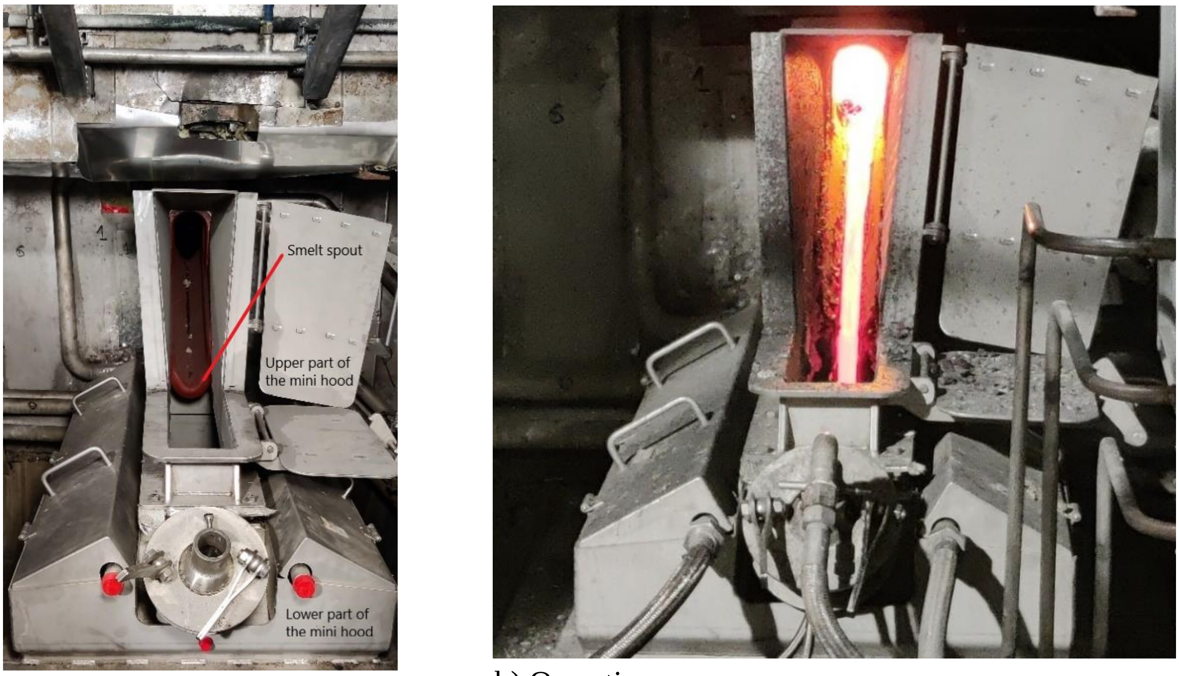


Figure 19. Chemical processes in the recovery boiler furnace (Vakkilainen 2017, p. 243).



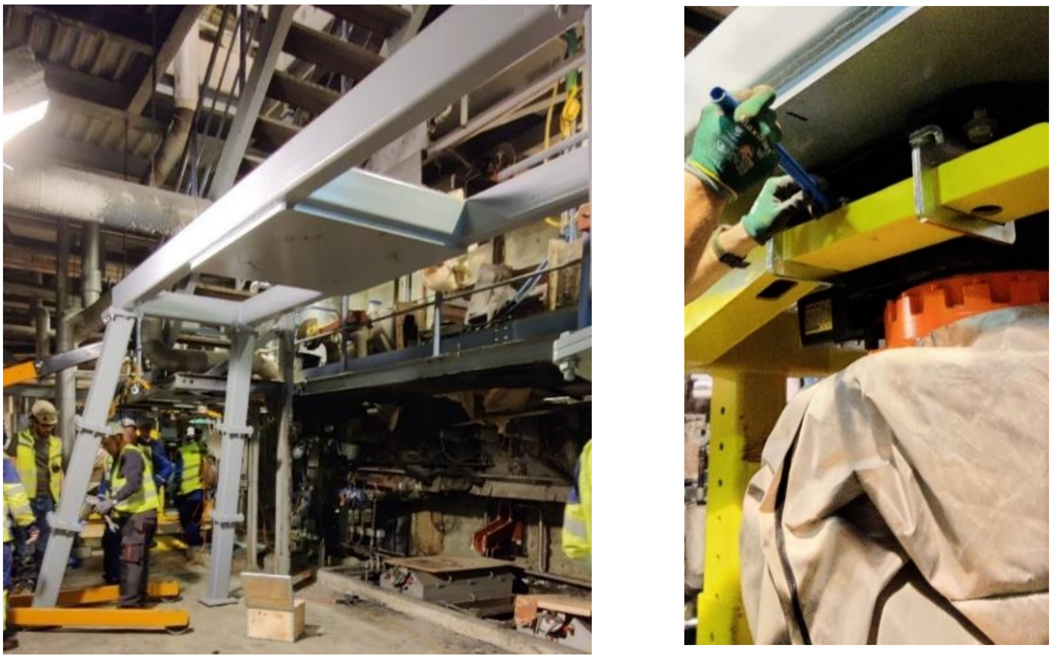
a) Before action.

b) On action.

Figure 20. An example structure of the smelt spout and hood.

The Smelt Spout Robot is always commissioned in advance in a test environment that mimics to the customer’s environment. The tests are important due to the variation of the robot’s environment. Size of the recovery boiler is varying and also the number of the smelt spouts and distance between them are varying. Thus, the movements of the robot are tested and adjusted in the test environment in advance. To facilitate the re-commissioning on site, it is especially important that the test environment includes all obstacles that may be affected the robot’s movements. The functionality of the robot programs is tested numerous times before the whole installation is transported to the final environment. The installation includes the robot and its controller, steel support structure, safety fences and the tool for the robot.

On the customer’s site, the package will be installed in the smelt spout area on the information on the drawings. The robot cell start-up begins with mechanical and electrical installations. First, the steel support structure (Figure 21, a) that includes four legs and the plate, are constructed utilizing a special designed lifting machine. Next, the location and stability of the steel support structure are ensured because they have a direct impact on the location of the robot and thus the location of the world coordinate system of the robot. After the mechanical installation of the steel support structure, the robot will be attached to it as is shown in the Figure 21, b.



a) The steel support structure consists of plate and four legs.

b) The robot is attached to the steel support structure.

Figure 21. A steel support structure for the Smelt Spout Robot.

Finally, a protective fence is built around the robot and the robot will be covered with a protective suit (Andritz 2019). The installation would also contain additional features such as a cleaning machine of the tool and a reduction sample box depending on the project. After the mechanical and electrical installations, the re-commissioning of the robot can be started. The robot is jogged to the service position where the cleaning tool is attached to the end of the robot arm. Then the robot expert starts to test and adjust the robot movements. Usually, some modifications are needed due to the mechanical differences between the test environment and customer’s environment.

The number of robot programs depends on the features of the robot. Currently, there are 7 – 9 different programs where the points should be taught. Most of the programs include large transition movements which are determined with respect to the world coordinate system. Usually, these programs are checked very quickly. The most time-consuming part is the commissioning of the cleaning programs because they include a lot of accurate movements which are sensitive to mechanical differences.

The cleaning programs are determined with respect to the base coordinate system, which is taught in the lower right corner of the smelt spout (Figure 22). Due to the there is no exact point for the origin, the location of the origin may differ from the test environment. Because the cleaning points are determined with respect to the base coordinate system, the difference can cause additional teaching work if the locations of the cleaning points are not correct. The inaccuracy of the base coordinate system is increased due to its location. If the robot is commissioned while the boiler is running (Figure 20, b), it is difficult to see the surface of the spout. Even in the situation of Figure 20, a, teaching can be difficult due to the darkness inside to the hood.

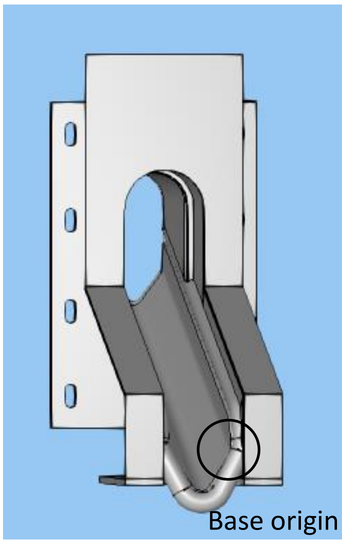


Figure 22. The origin of the base coordinate system is on the surface of the spout.

It is difficult to determine the same orientation for the smelt spout in the test environment as it really is. In addition, the orientation of the smelt spout depends on the recovery boiler and the heat movement. Therefore, the commissioning of the cleaning programs is started by evaluating the directions of the movements inside the hood. If the tool does not go straight on the desired line, the Euler Angles of the base coordinate system is modified by the robot expert who is evaluating the error. The evaluation can increase the error of the base coordinate system and it is also time-consuming.

The cleaning programs have been created utilizing traditional robot programming methods. They include a total of ten points that should be taught by manipulating the robot. The location of these points should be verified, due to the actual cleaning movements are defined relative to these points as relative motion commands. Due to the hoods and spouts are bolted together, there may be mechanical differences between the smelt spouts. Therefore, the starting points for the hood cleaning should be taught to each smelt spout separately.

The programs are divided into separate phases. A spout cleaning program contains a total of four different steps. First, the blockages from the beginning of the spout are removed before cleaning the surface of the spout with two different angles. In the last step, the tool is pushed into the recovery boiler to remove the solidified smelt between the spout and recovery boiler. A hood cleaning program is divided into three steps on each side. The right side of the hood is always cleaned first, starting with the roof cleaning, and ending with the bottom wall cleaning. The side wall is cleaned between the roof and the bottom wall. Finally, the same sequence of cleaning steps is repeated on the left side.

Currently, the right and middle smelt spouts are cleaned using the same orientation of the tool coordinate system. Due to the limitations of the robot axes, the tool coordinate system is rotated for the left smelt spout. In practise, the wrist of the robot arm is rotated as visualized in Appendix 4. However, one time-consuming step during the commissioning can be the middle hood (Figure 23) because it is located on the Y axis of the world coordinate system. Due to the limitations of the A5 axis (Figure 8), the teaching of cleaning points may require adjustment.

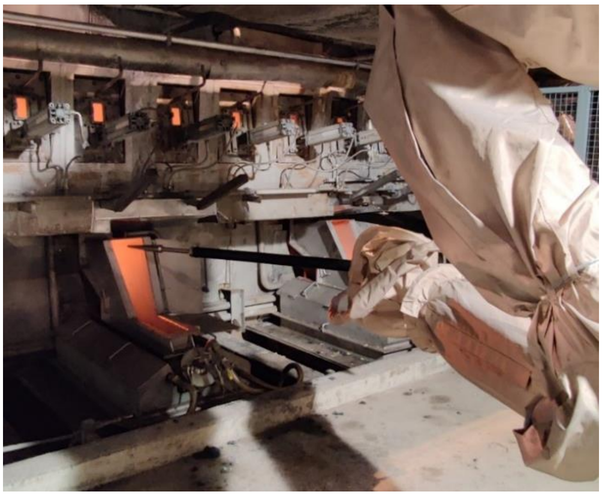


Figure 23. A commissioning of the hood cleaning program can be time-consuming for the middle smelt spout due to the limits of the robot's axes and the location of the smelt spout.

The final step of the commissioning is the cleaning adjustments. Due to the real performance of the robot can be seen when the melt is flowing and solidifying to the spout, the cleaning should be adjusted based on the cleaning quality. Currently, the cleaning parameters should be modified inside the relative motion commands. The commissioner should know the consequences of the modifications because the relationships of the parameter changes have not been considered in the code of the program. In practise, one modification can also require modifications elsewhere.

The positions of the spouts can be change as a function of time due to the heat movement of the recovery boiler. Thus, the movement of the boiler is measured in X, Y, Z directions using the digital sensors. The measured values are used to calculate the new locations of the base coordinate systems. To avoid the collisions, the location of the base coordinate system is always checked and redefined before the cleaning programs. The Table 7 expresses an example of the displacement of the recovery boiler. Initial values were measured when the boiler is cold due to the shutdown. Values describe the real time values after four days of start-up.

Table 7. An example displacement of the recovery boiler. The initial values were measured when the boiled was cold and real time values were measured after four days of start-up.

Sensor	Value (mm)	Initial value (mm)	Difference (mm)
X	162	194	-32
Y	255	242	13
Z	353	543	-190

As can be noticed from the Table 7, the recovery boiler has almost landed up to 200 mm, and it has also moved in X and Y directions. Based on the difference between the initial value and the real time value, the new locations of the base coordinate systems are calculated.

As was expressed earlier, the number of programs depends on the features of the robot. One of the mandatory programs is a tool straightness check where the tool is jogged to a specific point and its straightness is checked mechanically. The straightness of the tool should be checked regularly that the functionality of the cleaning process can be ensured. The other possible programs are a tool cleaning program and a reduction sample. In the tool cleaning program, the tool is cleaned (Figure 24) with a special designed cleaning machine because the melt also solidifies on its surface. In the reduction sample program, the robot takes a sample of the melt flow and takes it to be photographed to the reduction sample box. The reduction sample is used to measure the performance of the recovery boiler. These other programs are executed mainly in the world coordinate system because they consist of the large transition movements. There are also points that require precision. These points are also defined relative to the world coordinate system and therefore usually need to be re-taught on site.

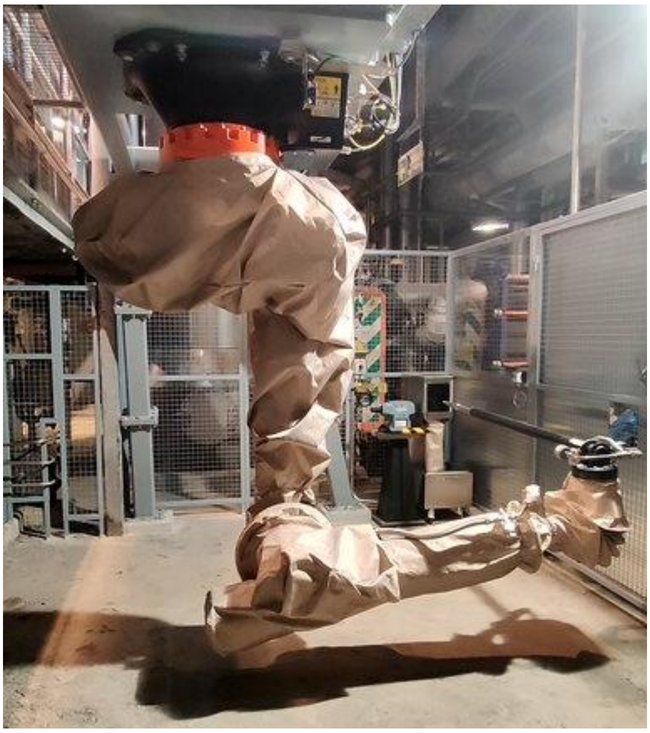


Figure 24. A tool cleaning machine removes the solidified melt on the surface of the tool.

The commissioning of the Smelt Spout Robot requires careful planning and testing, because there are a lot of affecting aspects. Mechanical installation failures such as the orientation and stability error of the steel support structure have a direct effect on the position of the robot. The location of the world coordinate system of the robot is therefore also moving relative to the spouts. Hence, the location of the base coordinate system needs to be adjusted and the locations of the points which are defined relative to the world coordinate system are changing. Normally, the minor changes in the large transition movements do not matter, but in a cramped environment these changes can cause additional teaching work.

There are several areas to expedite and facilitate the commissioning of the Smelt Spout Robot. First, the accuracy of the base coordinate systems should be increased by determining them to the specific points. They should be easily re-taught in the new environment. The certain location of the base coordinate system enables the parametrization relative to it. Therefore, there is no need to re-teach the cleaning points anymore. Due to the evaluation of the orientation of the smelt spout is time-consuming, the orientation calculation should be created to determine the exact orientation of the base coordinate system.

Even if the base coordinate systems and the cleaning point were taught perfectly, the parameters of the cleaning movements must usually be changed due to mechanical differences. Currently, the modifications in the cleaning programs require a lot of understanding about the robotics, the directions of the coordinate systems and the structure of the program. The cleaning points can only be re-taught by jogging the robot to the desired location or changing the parameters in a DAT -file. The parameters of the actual cleaning movements can only be modified inside the relative motion commands thus increasing the risk of making mistakes.

In the current programs the effect of parameters changes should be understood. For example, if the cleaning angle is needed to change, there is a total of three steps should be made. First, the approach points to which the relative movements are related should be re-taught with a new cleaning angle by jogging the robot. Then, the length of the actual cleaning movement should be adjusted inside the relative motion command. Finally, the program should be executed and check whether the number of cleaning steps should also be changed. Therefore, the parameter changing should be simplified to considering the relationship between them automatically.

3.2 Creating the simulation environment

The development process started by creating the testing environment in the simulation according to the instructions that were introduced in the Chapter 2.2. It included the steel support structure, robot, cleaning tool and three smelt spouts as is expressed in the Figure 25. The 3D models of the hoods were the updated versions because they will be used in the next projects.

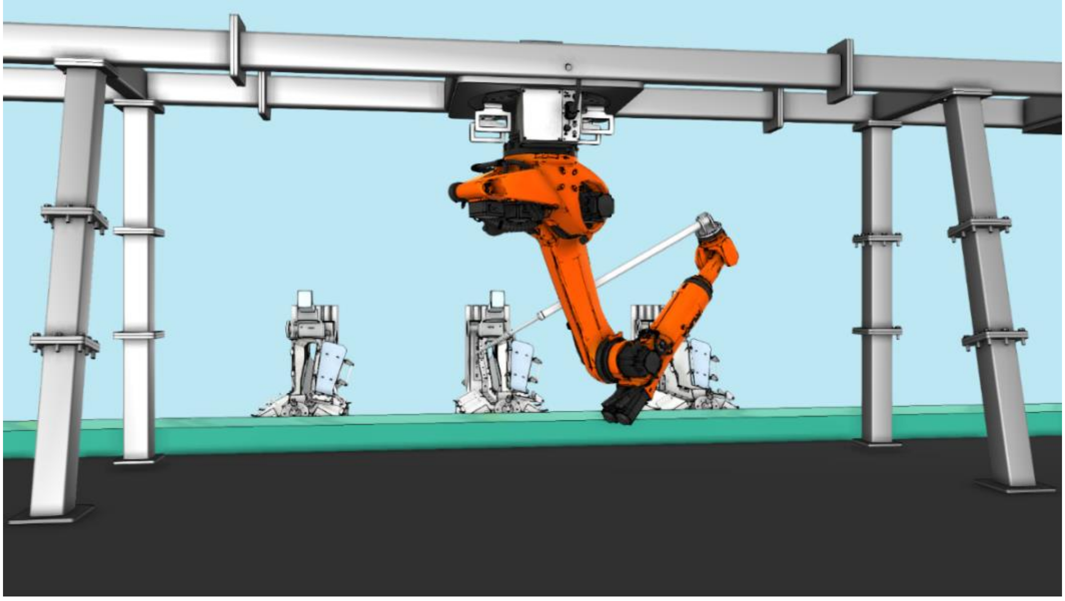


Figure 25. A simulation environment of the smelt spout area.

3.3 Development of the teaching process of the base coordinate systems

The first development step was to create a method to optimize the base coordinate system teaching process. As is expressed earlier there was not a specific point for the base coordinate system, and its orientation was determined by evaluating the orientation according to the cleaning movements. In the new teaching method, the base coordinate system was defined at a certain point that can be seen visually and thus easily measurable by a real robot. Its location was determined on the hood surface that it can be taught safely when the door of the hood is closed.

The determining the orientation of the base coordinate system is not anymore based on the estimations because in the new method the rotations can be calculated by measuring a total of three additional points: **P**$_1$, **P**$_2$ and **P**$_3$. The **P**$_1$ is representing the calculation origin. The point **P**$_2$ should be measured along +X axis and **P**$_3$ should be in a positive XY plane with respect to the calculation origin, as is shown in the Figure 26. The actual calculation program was found on the Internet and applied to the intended use. It is based on the 3-point optimization method of the base coordinate system. Although it was not an official version of the 3-point optimization method, its functionality was verified from the KUKA Global Customer Service.

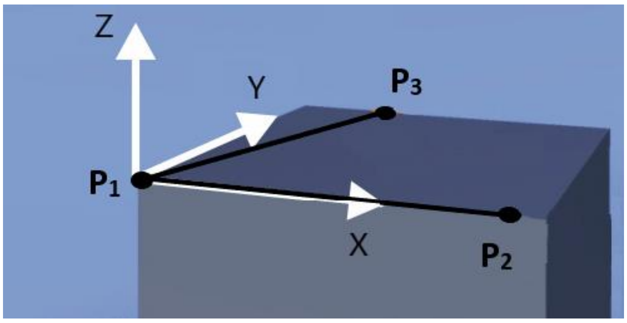


Figure 26. The orientation of the hood and spout can be calculated by measuring three additional points: **P**$_1$, **P**$_2$ and **P**$_3$.

In the new teaching process, the robot is automatically manipulated close to the smelt spouts using the pre-programmed movements. In addition to the automatic approach, there are two ways to measure the points. First, the robot can be jogged to the specific points using the robot teach-pendant. This method was designed for people who have experience in robotics and know how to jog it. In the second way, the robot can be manipulated to the points by measuring the distance between the desired point and the TCP. By writing the dimensions from the measurements to the new control parameters (Table 8), the location of the TCP can be moved along X, Y and Z axis about the world coordinate system. The depth can be adjusted by moving the tool along the X axis of the tool coordinate system. The second way was created for people who do not have the experience in the robotics.

Table 8. Parameters for the base teaching process.

Parameter	Description
Move X	TCP can be moved along the X axis of the world coordinate system.
Move Y	TCP can be moved along the Y axis of the world coordinate system.
Move Z	TCP can be moved along the Z axis of the world coordinate system.
Move Depth	TCP can be moved along the X axis of the tool coordinate system.

When the measurement point is reached, its parameters are automatically stored in the FRAME variable after user approval. The parameters are collected by utilizing $POS_ACT system variable by storing the location of the tool coordinate system with respect to the world coordinate system. Once a total of 3 points have been measured for the calculation, the base coordinate system is automatically updated to the desired point with the calculated Euler Angles. The calculation was based on the utilization of the rotation matrix. The vector between the points **P₂** and **P₁**, was determined as follows:

$$\mathbf{P_{21}} = \left(X_{\boldsymbol{P}_2} - X_{\boldsymbol{P}_1}\right)\mathbf{i} + \left(Y_{\boldsymbol{P}_2} - Y_{\boldsymbol{P}_1}\right)\mathbf{j} + \left(Z_{\boldsymbol{P}_2} - Z_{\boldsymbol{P}_1}\right)\mathbf{k} \tag{3.1}$$

and the vector between the points **P₃** and **P₁** was written as:

$$\mathbf{P_{31}} = \left(X_{\boldsymbol{P}_3} - X_{\boldsymbol{P}_1}\right)\mathbf{i} + \left(Y_{\boldsymbol{P}_3} - Y_{\boldsymbol{P}_1}\right)\mathbf{j} + \left(Z_{\boldsymbol{P}_3} - Z_{\boldsymbol{P}_1}\right)\mathbf{k} \tag{3.2}$$

where, $X_{\mathbf{P}_{1-3}}$, $Y_{\mathbf{P}_{1-3}}$ and $Z_{\mathbf{P}_{1-3}}$ are the components of the vectors relative to the world coordinate system. These vectors were normalized by dividing them by their length to obtain unit vectors as follows:

$$\mathbf{P_X} = \frac{\mathbf{P_{21}}}{|\mathbf{P_{21}}|} \tag{3.3}$$

$$\mathbf{P_{XY}} = \frac{\mathbf{P_{31}}}{|\mathbf{P_{31}}|} \tag{3.4}$$

By calculating the cross product of the vectors $\mathbf{P_X}$ and $\mathbf{P_{XY}}$, the normal vector $\mathbf{P_Z}$ that is perpendicular to the vectors $\mathbf{P_X}$ and $\mathbf{P_{XY}}$ was determined as follows:

$$\mathbf{P_Z} = \mathbf{P_X} \times \mathbf{P_{XY}} \tag{3.5}$$

When calculating the cross product of the vectors $\mathbf{P_X}$ and $\mathbf{P_Z}$, the perpendicular normal vector $\mathbf{P_Y}$ can be defined as:

$$\mathbf{P_Y} = \mathbf{P_Z} \times \mathbf{P_X} \tag{3.6}$$

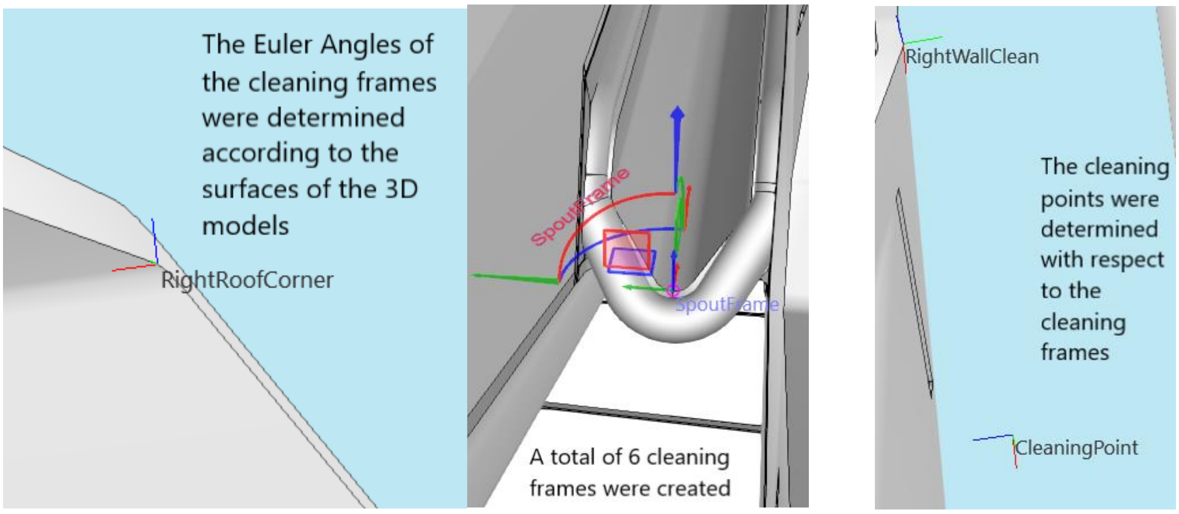
Finally, there will be three unit-vectors $\mathbf{P_X}$, $\mathbf{P_Y}$ and $\mathbf{P_Z}$ which are perpendicular to the each other and they are representing the coordinate system that is defined by utilizing the orientation of the hood. Therefore, the rotation of the base coordinate system can be defined by calculating the Euler Angles of the determined coordinate system. As is expressed earlier, KUKA is using ZYX Euler convention and therefore the rotation matrix, **A**, for the three successive rotations can be expressed as was shown in the Equation 2.7. And thus, the Euler Angles can be calculated from the Equations 2.9 – 2.16.

3.3 Parametrization of the cleaning programs

The following development steps were to parametrize the spout and hood cleaning programs. As was expressed earlier, the old cleaning programs were built using traditional programming methods. They were based on the relative motion programming, and the cleaning points that were defined by a real robot. The modifications required an understanding of the program itself, and the parameters changing was difficult because the actual cleaning movements had to be modified inside the relative motion commands.

The parametrization started by creating additional frames that were determined to different points of the hood and spout. A total of five additional frames were defined for the hood cleaning and one for the spout cleaning. These frames were determined as an offset from the origin of the base coordinate system. The shapes of the 3D models were used to determine the orientation of the frames as is shown in the Figure 27, a.

The cleaning programs changed completely because all the cleaning points were converted to POS variables. Also, the actual cleaning movements (60) were converted from the relative motion commands to the position variables and SLIN commands. Therefore, a total of 24 cleaning points were determined and all of them were defined with respect to some additional frame (Figure 27, b) depending on the cleaning phase. Only two commands were left as relative motion commands. They were previously defined as SLIN_REL #TOOL commands, but due to bugs in the simulation software, they were converted to LIN_REL commands to perform the same motion. Instructions for the modifications were obtained from KUKA Global Customer Service. According to them, the bug was related to spline motions, and it will be fixed in the next update.



The Euler Angles of the cleaning frames were determined according to the surfaces of the 3D models

A total of 6 cleaning frames were created

The cleaning points were determined with respect to the cleaning frames

a) Additional frames were created for the parametrization.

b) Visualized cleaning point.

Figure 27. Cleaning points were determined with respect to the new additional frames.

These changes can be considered significant because they made the cleaning movements parametrizable. The parameters of the position variables can be determined anywhere in the program, therefore almost 30 control parameters were created. Because the points were determined with respect to the additional frames, their parameters can be adjusted along the axes of these frames. Due to the orientation of the frames was determined based on the shapes of the objects, the parameters of the cleaning points can be adjusted along the surfaces of the spout and hood. The control parameters were divided into different cleaning phases. Therefore, some of them were determined multiple times (Table 9) that the cleaning process can be controlled in the smaller parts.

Table 9. A total of 30 control parameters were created to adjust the cleaning programs.

Cleaning parameter	Number of parameters
Width	2
Depth	9
Angle	4
Number of steps	2
Step length	4
Side	2
Tuning parameters	2
Cleaning length	2

The values of the control parameters can be modified at the beginning of the program, therefore there is not anymore need to find a specific motion command to modify the cleaning process. The control parameters are used to modify the parameters of the cleaning points, for example, as follows:

$$\mathrm{CleaningPoint.Y} = \mathrm{CleaningPoint.Y} - \mathrm{Width} \tag{3.7}$$

$$\mathrm{CleaningPoint.Z} = \mathrm{CleaningPoint.Z} - \mathrm{Depth} \tag{3.8}$$

Because one of the purposes of this research was to decrease the level of competence required during the commissioning, the relationships between the parameter changes were also considered. In practice, one parameter change is not anymore requiring the changes elsewhere. This was implemented by creating mathematical equations between the parameters and paying attention to their order. Thus, the required changes to the other parameters are made automatically. For reasons of secrecy, these mathematical equations have been excluded from the public research paper. In any case, a couple of examples of the parameter change automation are shared. Due to the shape of the tool, the cleaning depth is automatically updated if the cleaning width is modified as is expressed in the Figure 28. In the previous programs, both had to be modified separately. Another example of the benefits of the new programs is the cleaning angle modification, where the robot commissioner can set a new cleaning angle, and a new cleaning point is calculated automatically in the background of the program. In the previous programs, the angle change required at least three additional changes.

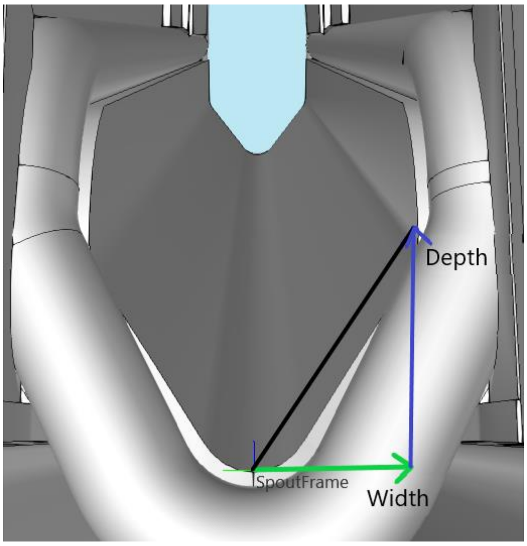
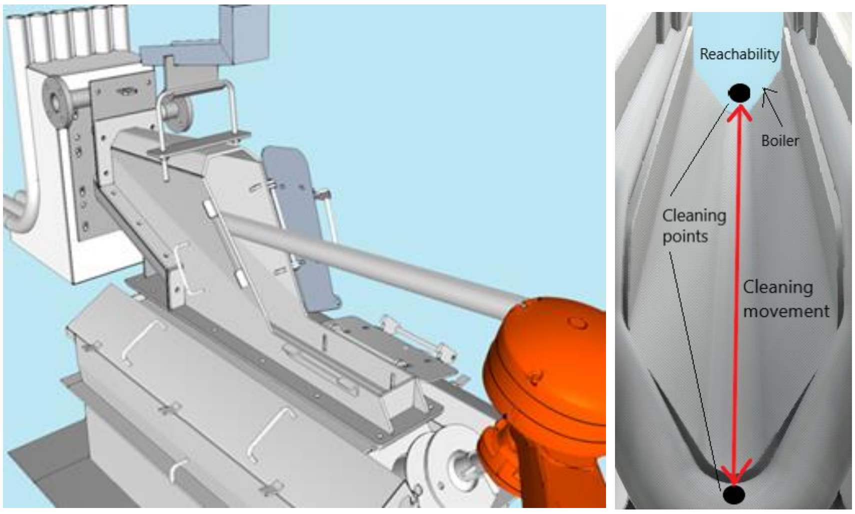


Figure 28. A cleaning depth is automatically adjusted when the cleaning width is changed due to the shape of the spout.

Some restrictions were also needed to automate parameter changes. As can be seen from the Figure 29, a, the roof of the hood restricts the cleaning movements, thus the maximum cleaning trip must be calculated after the parameter changes. The maximum cleaning trip depends on the angle between the spout and the tool; thus, the limiting point was defined for the cleaning. The maximum cleaning trip will be automatically calculated based on the determined cleaning angle and the limiting point. Due to the actual cleaning trip depends on the location of the starting point, the number of steps and the step length, the maximum steps were calculated based on the maximum cleaning trip. Therefore, the cleaning movements will be executed based on the maximum steps and the collisions with the hood roof can be avoided. Thus, these limitations reduced the possibility of human error and expedited the commissioning of the robot.

Although the parameters were created to control the cleaning programs, they can be also utilized in the future to select the correct size of the robot and thus to optimize the robot cell. As was expressed earlier, the size of the recovery boiler also affects the number of spouts and the distance between them. Therefore, the parameter called “Reachability” was created (Figure 29, b). It can be used to test the reachability of the robot in the different environments in the future. In practice, it is possible to test how far the robot tool extends inside the boiler, which is one of the decisive factors in choosing a robot. These reachability tests can be executed already in the design stage by using the robotic simulation. Thus, the suitability of the robot can be also ensured in the new scenarios.



a) A roof of the hood restricts the cleaning.

b) Reachability of the robot can be tested.

Figure 29. New control parameters are used to calculate the maximum cleaning trip. One of the new parameters can be used to evaluate the reachability of the robot.

As previously stated, there may be mechanical installation errors between the spout and hood due to the attachment between them. Since the base coordinate system will be determined by the points of the hood, the mechanical error should be correlated in the spout cleaning program. One of the new control parameters was created to change the location the spout cleaning frame along the Y axis. The adjustment will be performed using the geometric operator as follows:

$$\mathrm{CleaningFrame.Y} = \mathrm{CleaningFrame.Y} : \mathrm{Tuning} \tag{3.9}$$

The mathematical background of the geometric operator was expressed in the Chapter 2.1. The location of the cleaning frame can be adjusted for each spout separately.

3.4 Optimization of the tool orientation

In the old versions of the cleaning programs, the actual cleaning movements were executed using the directions of the tool coordinate system in the relative motion commands. Due to the carrot-like shape of the tool, the orientation of the tool relative to the hood wall could not be optimized in the relative motion commands. In practice, only a certain part of the tool cleaned the wall surfaces. In the new cleaning programs, where the cleaning points are determined with respect to the additional frames, the optimization of the tool orientation was possible. The orientation of the tool was optimized according to the orientation of the wall and thus the cleaning movements will be executed between the cleaning points using SLIN commands and the directions of the frames. As a result, the cleaning will be smoother in the future.

As was expressed earlier, the commissioning of the middle hood cleaning program was time-consuming due to the limits of the A5 axis and thus the cleaning points had to be modified several times. Due to the optimization of the tool orientation, there would have been more problems with the A5 axis in the new version of the program. Therefore, a new additional rotation feature was created where the middle hood can be cleaned up using two different orientations of the tool coordinate system (Figure 30). This gives the A5 axis considerably more room to maneuver depending on the situation. The new feature can be utilized on the left and right sides of the hood as well as for the spout cleaning.

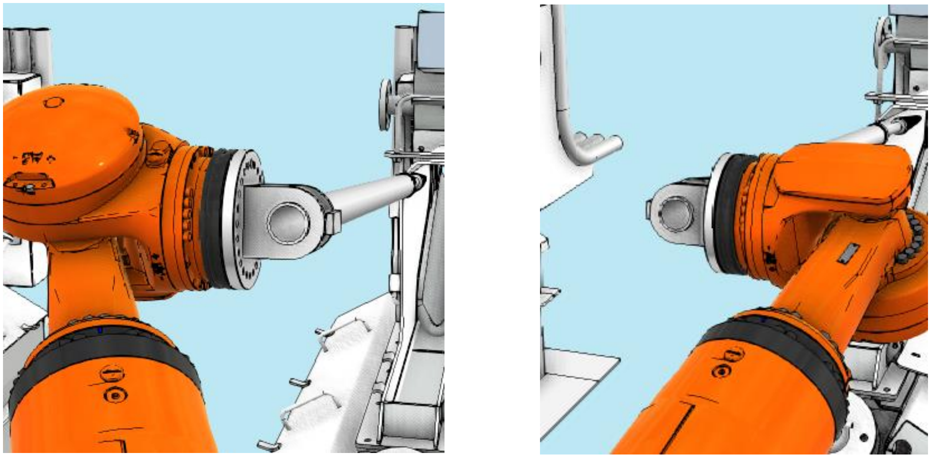


Figure 30. The middle hood and smelt spout can be cleaned using different orientations of the tool coordinate system.

The axes' positions during the cleaning programs depend on many aspects, such as the cleaning angle, the location of the middle spout relative to the robot and orientation of the smelt spout. Because all of them can change, it is difficult to determine the exact point for the rotation. Thus, the limits of rotation feature can be roughly determined based on the locations of the cleaning frames in the ideal situation where the middle spout is in the same line with the robot. The location of center of the spout is always known with respect to the base coordinate system due to the parametrized frame that was expressed in the Figure 27, a. Hence, its location with respect to the world coordinate system can be obtained by the geometric operator as:

$$\mathrm{SpoutLocation} = \mathrm{BASE\_DATA[2]}:\mathrm{SpoutFrame} \tag{3.10}$$

The rotation limit point for the spout cleaning can be roughly determined to be zero. In practice, the tool orientation is rotated if the Y coordinate of the cleaning frame is greater than zero because then the mobility of the A5 axis can be improved by the rotation feature. When cleaning the left wall (LW) of the hood, the A5 axis is even closer to its limit than it was before due to the optimized tool orientation. On the right wall (RW), the optimization made more space for the A5 axis. Due to the opening shape of the hood and the limitations of the robot axes, each side must have its own limit point for the rotation. The corners of the hood walls were parametrized with respect to the base coordinate system; therefore, their locations can be calculated with respect to the world coordinate system as:

$$\mathrm{RightCornerLocation} = \mathrm{BASE\_DATA}[2]\colon \mathrm{RightCorner} \tag{3.11}$$

$$\mathrm{LeftCornerLocation} = \mathrm{BASE\_DATA}[2]\colon \mathrm{LeftCorner} \tag{3.12}$$

From the Equations 3.11 and 3.12, the locations of the corners in the ideal situation can be obtained. Based on these ideal locations, the rotation limits can be determined as is expressed in the Table 10. In practice, the right side of the hood is always cleaned in the traditional way, unless the location of the right corner exceeds the X axis of the world coordinate system. The left side of the hood is always cleaned up using the different orientation of the tool coordinate system unless the location of the left corner is less than or equal to zero. It is important to note that these limits are recommendations based on the ideal situation. Actual limits should be determined on a case-by-case basis.

Table 10. Recommendations points for the rotation.

Cleaning phase	**Y- coordinate of the rotation point with respect the world**
Spout Cleaning	SpoutLocation.Y > 0
Hood Cleaning – RW	RightCornerLocation.Y <= 0
Hood Cleaning – LW	LeftCornerLocation.Y => 0

Due to the new rotation feature, the additional program was created around the cleaning programs. Its purpose is to check the locations of the relevant points always before the cleaning programs. If the conditions (Table 10) associated with the cleaning programs are met, the tool is rotated before the cleaning in front of the smelt spout. If the orientation is changed, the orientations of the cleaning points are automatically updated to the correct value. Although the rotation feature is only needed at the middle spout, the orientation check program is executed before each spout. In this way it is possible to update the orientation of the cleaning points also to the left spout where the tool orientation differs from the right spout as was expressed in the Appendix 4.

3.5 Practical tests

After the development process, the functionality of the new programs was tested in the test environment (Figure 31). The test environment included a steel support structure, three smelt spouts and the robot which model was KR250 R2700-2 C. The hood models were still the old ones (Figure 32) because the new models were not still arrived at the testing site.

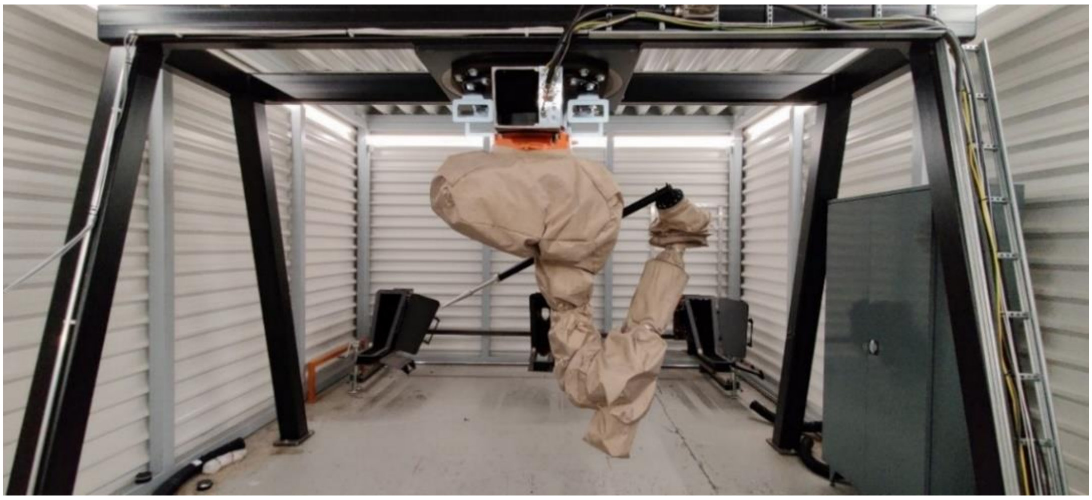


Figure 31. Practical tests were performed in the test environment that included a steel support structure, three smelt spouts, and the robot.

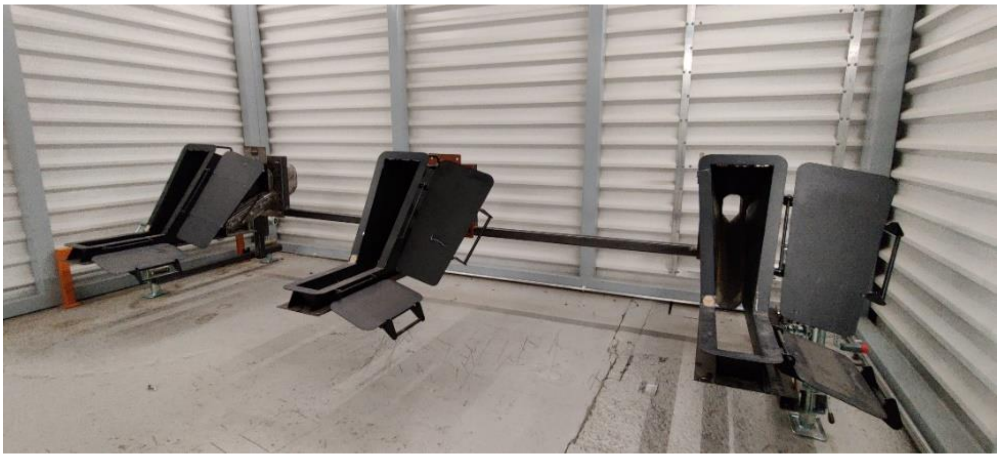
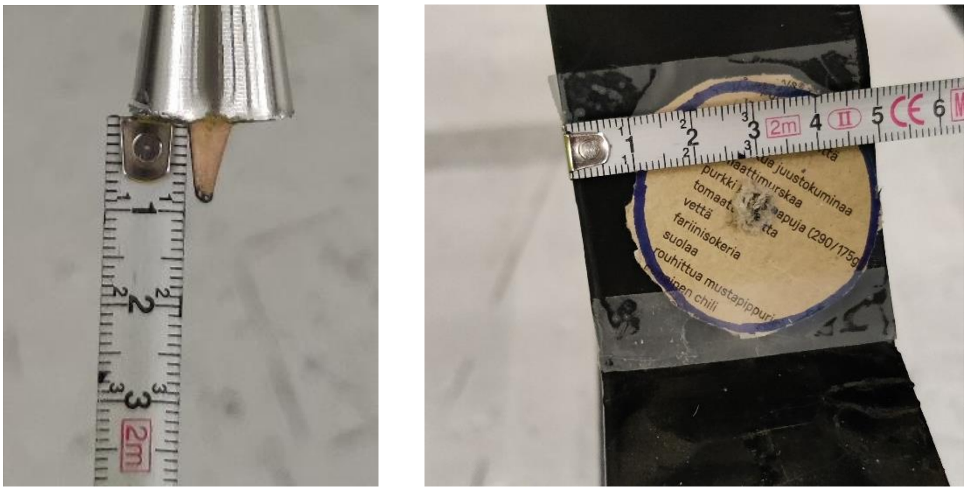


Figure 32. The hood models had not been updated in the test environment.

The practical tests began by creating the teaching tool for the tests. A truncated pen was placed at the end of the cleaning tool, as is shown in the Figure 33, a. In this way, it was possible to measure the points more accurately. Due to the old hood models, the locations of the base coordinate systems had to be marked as is expressed in the Figure 33, b.



a) A teaching tool for the tests.

b) The locations of the base coordinate systems were marked on the old hood models.

Figure 33. A truncated pen was placed at the end of the cleaning tool to measure the locations of the base coordinate systems.

The practical tests were performed a total of three times. Each of tests began with teaching the base coordinate systems of the spouts using a new teaching method that was expressed in the Chapter 3.3. The new cleaning programs were then transported to the robot for the practical tests. The first tests were executed by the author of this research. He had intermediate knowledge of robotics because he understood the basics of robotics but had limited experience. The practical tests were his fourth time manipulating a real robot. The following tests were performed in collaboration with a robot expert. In the second and third tests, the purpose was also testing the automation alongside the new teaching method and new cleaning programs. For the automation, the robot expert had created PLC programs based on the development steps of this research.

The test environment remained the same in the tests, but there were some differences between the robot and the tool in the tests. These differences are expressed in the Table 11:

Table 11. The test environment remained the same in the practical test, but there were mechanical differences.

Test number	**Robot (KR250 R2700-2 C)**	**Tool**
1	• Old robot that was already used to execute the old versions of the programs.	• The tool was old and already used.
2	• The robot was replaced by a new one. • The model was same than in the first test.	• The tool was same than in the first test.
3	• Same robot than in the second test.	• The tool was replaced by a new unused tool.

The functionality of the new base teaching process was tested by manipulating the tool to the approach points of the smelt spouts using preprogrammed movements from the simulation environment. After the approach points, the robot was jogged to the measurement points by using the robot teach-pendant. The locations for the base coordinate systems were determined as was shown in the Figure 35, b. The points $\mathbf{P_1}$, $\mathbf{P_2}$ and $\mathbf{P_3}$ for the calculation were measured based on surface of the hood based on the instructions in the Chapter 3.3. However, there were no exact points for them.

Therefore, a total of four points were measured from each smelt spout as is expressed in the Table 12. The calculation points were collected only from the first and third practical tests because the tool was bent during the teaching process in the second tests. In any case, the calculation was executed successfully each time. The base coordinate systems with calculated Euler Angles are expressed in the Table 13.

Table 12. Coordinates of the measured points in the first and third practical test.

Spout 1	**Teaching process 1**	**Teaching process 3**
Origin	{X 2393.87, Y -1634.43, Z -1966.54}	{X 2390.52, Y -1630.23, Z -1975.64}
P_1	{X 1993.22, Y -1842.45, Z -1981.46}	{X 1999.51, Y -1839.30, Z -1989.58}
P_2	{X 2370.00, Y -1836.89, Z -1986.21}	{X 2371.85, Y -1833.91, Z -1992.72}
P_3	{X 2370.00, Y -1608.08, Z -1986.92}	{X 2373.75, Y -1605.01, Z -1994.37}
Spout 2		
Origin	{X 2393.99, Y 100.22, Z -1980.77}	{X 2391.04, Y 103.59, Z -1987.08}
P_1	{X 1981.78, Y -110.69, Z -2014.63}	{X 1991.41, Y -106.93, Z -2018.83}
P_2	{X 2367.32, Y -105.66, Z -2010.43}	{X 2366.77, Y -102.32, Z -2015.80}
P_3	{X 2367.32, Y 125.76, Z -2004.77}	{X 2354.81, Y 128.43, Z -2009.96}
Spout 3		
Origin	{X 2374.93, Y 1828.22, Z -1957.88}	{X 2380.19, Y 1825.26, Z -1951.04}
P_1	{X 1964.28, Y 1621.44, Z -1983.09}	{X 1976.75, Y 1617.79, Z -1979.42}
P_2	{X 2346.02, Y 1625.04, Z -1982.40}	{X 2345.21, Y 1620.51, Z -1979.78}
P_3	{X 2345.56, Y 1855.16, Z -1975.97}	{X 2327.27, Y 1851.55, Z -1973.95}

As was expressed earlier, the measured points were collected into the FRAME variables. Then the orientation calculation was executed in the robot controller. After the calculation, the base coordinate systems were automatically determined in the system.

Table 13. Calculated base coordinate systems for the spouts.

Base Coordinate Systems - Teaching process 1	
Spout 1	{X 2393.87, Y -1634.43, Z -1966.54, A 0.85, B 0.72, C -0.17}
Spout 2	{X 2393.99, Y 100.22, Z -1980.77, A 0.75, B -0.62, C 1.39}
Spout 3	{X 2374.93, Y 1828.22, Z -1957.88, A 0.54, B -0.10, C 1.60}
Base Coordinate Systems - Teaching process 2	
Spout 1	{X 2393.26, Y -1634.53, Z -1969.02, A 0.79, B 0.51, C -0.51}
Spout 2	{X 2393.48, Y 98.14, Z -1983.40, A 0.78, B -0.55, C 1.48}
Spout 3	{X 2378.54, Y 1830.75, Z -1956.16, A 0.58, B 0.11, C 1.80}
Base Coordinate Systems - Teaching process 3	
Spout 1	{X 2390.52, Y -1630.23, Z -1975.64, A 0.83, B 0.48, C -0.40}
Spout 2	{X 2391.04, Y 103.02, Z -1986.72, A 0.70, B -0.46, C 1.47}
Spout 3	{X 2380.19, Y -1825.26, Z -1951.04, A 0.42, B 0.06, C 1.44}

Thus, the teaching processes were successfully executed by two different persons. During the teaching processes, the functionality of the parametrized teaching process was also tested. There, the robot was manipulated to the measurement points by modifying the control parameters that were introduced in the Table 8. Due to the cramped environment, and the length and shape of the tool, also the B angle with respect to the world coordinate system was modified. Therefore, the knowledge of the teach-pendant was required.

As was expressed earlier, the first practical test was executed by the author of this research. After the base teaching process, the spout and hood cleaning programs were tested. They worked immediately as was expected, and there were no surprising movements. Some adjustments had to be done due to mechanical differences between the spouts. The cleaning was executed more on the left side at the first and the second smelt spout, as is expressed in the Table 14. Thus, the cleaning frame of the spout was moved along the negative Y axis by modifying the new control parameter (Equation 3.9). Specific modifications and results from the first practical tests are expressed in the Table 14.

Table 14. Summary of the results of the first practical tests.

Procedure	**Results**		
	Smelt Spout 1 - Right	**Smelt Spout 2 - Middle**	**Smelt Spout 3 - Left**
Spout Cleaning Hood Cleaning	• The tool cleaned the spout more from the left side. • Tool touched the spout slightly on the right at the beginning and on the left at the end.	• The tool cleaned the spout more from the left side. • The cleaning points were higher at the end of the spout.	• Programs were executed successfully.
Modifications	• Cleaning frame of the spout was moved -7.5 mm along –Y axis.	• Cleaning frame of the spout was moved -5 mm along –Y axis. • The B-angle of the base coordinate system was increased.	• No modifications.

The second tests were performed by a robot expert, under the supervision of the author of this research. The purpose of the tests was to test the automation along the new methods and programs. During the automation test, the tool hit the post of the support structure and was bent. The collision occurred after the first base coordinate system had already been taught. Therefore, the base coordinate systems for the middle and left spout were taught with an indirect tool. However, the required modifications were similar to the first test, as is shown in the Table 15.

The third test was also executed in collaboration with a robot expert. The tool was replaced by a new one as was expressed in the Table 11. The results of the final practical tests corresponded well to previous tests and the required modifications were similar. In addition to the previous modifications, the A angle of the base coordinate system for the first spout was modified because the tool did not move along the X axis of the cleaning frame. The cleaning points were more on the left at the end than at the beginning. The summary of the results is expressed in the Table 16.

Table 15. Summary of the results of the second practical tests.

Procedure	**Results**		
	Spout 1 - Right	**Spout 2 - Middle**	**Spout 3 - Left**
Spout Cleaning	• The tool cleaned the spout more from the left side. • Tool touched the spout slightly on the right at the beginning and on the left at the end.	• The cleaning points were higher at the end than at the beginning.	• Program was executed successfully.
Modifications	• Cleaning frame of the spout was moved -7.5 mm along –Y axis.	• The B-angle of the base coordinate system was increased 0.5 degrees.	• No modifications.

Table 16. Summary of the results of the third practical tests.

Procedure	**Results**		
	Spout 1 - Right	**Spout 2 - Middle**	**Spout 3 - Left**
Spout Cleaning	• The tool cleaned the spout more from the left side. • The tool did not move directly along the X axis of the cleaning frame. The difference was approximately 5 mm at the end.	• The tool cleaned the spout more from the left side. • The cleaning points were slightly higher at the end than at the beginning.	• Program was executed successfully.
Modifications	• Cleaning frame of the spout was moved -5 mm along –Y axis. • The A-angle of the base coordinate system was increased 0.4 degrees.	• The B-angle of the base coordinate system was increased 0.3 degrees. • Cleaning frame of the spout was moved -6 mm along –Y axis.	• No modifications.

Due to the parametrized robot programs, the robot no longer had to be manipulated during the commissioning of the cleaning programs. As was shown in the Tables 14 – 16, the programs were executed with small modifications. The functionality of the control parameters was also tested successfully in the practical tests. They worked as was expected, so the cleaning process could be further adjusted by changing the values of the control parameters such a cleaning angle, width, and depth.

The new cleaning programs were also connected to the PLC by the robot expert. Thus, the parameters can be easily changed by written their values on the touch screen. The limit values were also determined to the parameters. The detailed instructions about the parameters changes and new features were also written but were omitted from the public version of the research as was expressed earlier.

4. Result analysis

The results of the research were analyzed in this chapter. The chapter summarizes the main development steps and evaluates their success by analyzing the accuracy of the practical tests. In addition, the utilization of the simulation is reviewed and the recommendations for the future are expressed.

4.1 Summary of the development process

The purpose of the research was to facilitate and simplify the commissioning of the Smelt Spout Robot. The purpose was also decreasing the required level during the robot commissioning by parametrizing the robot programs. The old programs had been created using a traditional robot programming which made commissioning challenging and time-consuming. Due to the growing order backlog, commissioning had to be accelerated and simplified to anticipate future challenges. This chapter summarizes the steps in the development process by analyzing the modifications.

One of the drawbacks previously was the inaccurate location of the base coordinate system. The inaccuracy was also increased by evaluating the orientation of it. If the locations of the base coordinate systems on site differed from the test environment, this could be noticed as a difference in the cleaning points that were determined as an offset from the origin. Thus, these points had to be re-taught on site, which increased the time required for the commissioning. In addition, it also increased the complexity of the commissioning because the points had to be re-taught by manipulating the tool of the robot using a teach-pendant.

The origin of the base coordinate system was previously located inside the hood on the spout’s surface, making it difficult to determine. As was expressed in the Chapter 3.3, at the beginning of the development process the origin of the base coordinate system was determined to the specific point on the surface of the hood. By visualizing the location of the origin, its teaching accuracy was improved. In addition, it can be determined quickly in the future when its location is clearly known. Due to the origin was determined outside of the hood, the safety of the teaching situation was also improved. The closed door protects the commissioner from sudden splashes from the boiler if the boiler is running.

The level of competence required during the commissioning was not significantly reduced by determining the location of the origin of the base coordinate system to the exact point because its orientation is varying between the spouts. Therefore, the aim was also to get rid of the evaluation of the orientation of the base coordinate system. As was expressed in the Chapter 2.1.1, the orientation of a rigid body can be determined by the Euler Angles. Understanding them requires knowledge of the mathematical background of the rotations, as the rotations can be determined in different orders. Thus, it can be difficult for the human brain to understand the significance of the rotation angles.

Hence, the orientation calculation was integrated into the teaching process of the base coordinate system, as was expressed in the Chapter 3.3. For the calculation, the robot commissioner must measure three additional points to calculate the Euler Angles. After the calculation, the Euler Angles are automatically determined to the base coordinate system. Thus, the benefit of the calculation is that the Euler Angles do not to be evaluated anymore. On the other hand, the disadvantage of the calculation would be increased number of the point to be taught. However, it can be assumed that is easier to measure the locations of three additional points after measuring the location of the origin of the base coordinate system than to start estimating the Euler Angles based on the robot movements.

In any case, manipulating the robot to a few extra points is easy for a robot expert, but for the inexperienced this can cause difficulties. Therefore, an automated program was created in which the robot can be manipulated by changing the values of the parameters (Table 8). This way, the commissioner does only need to know the directions of the axes of the world coordinate system. The teaching process was further facilitated in collaboration with the robot expert, who connected it to the PLC. Thus, the robot can be manipulated from the touch screen by pressing the move buttons that were based on the parameters of the Table 8. The comparison between old and new teaching process is expressed in the Table 17.

Table 17. Comparison of the old and new teaching process for the base coordinate systems.

Teaching process	**Old**	**New**
Exact point for the origin	No	Yes
Location of the base coordinate system	Inside the hood	Outside of the hood
Rotations	Evaluated	Calculated
Teaching points	1	4
Automated	No	Yes

Even if the number of teaching points increased, the benefits of the new teaching process can be considered significant. By increasing the accuracy of the base coordinate system, the error of the cleaning points is reduced. In this way, the offset of the points from the origin does not need to be adjusted and the additional teaching work is avoided.

One of the main purposes of the research was to parameterize the robot programs using robotic simulation. Because the purpose of the parametrization was to expedite the commissioning and decrease the level of competence, it was also important to get rid of the problem areas. Previously the cleaning points had to be taught by manipulating the real robot to the cleaning points. Thus, the accuracy of the points depended on the level of expertise of the commissioner. These points were very important for the cleaning process as they were the starting points for the actual cleaning movements. In practice, the quality of the cleaning programs depended also on the level of expertise of the commissioner.

Therefore, there were significant changes in the cleaning programs. The advantages of the simulation were utilized in the editing of the programs by creating additional frames for the cleaning programs based on the 3D models of the smelt spout and hood. The frames had not been used in the previous programs because the determination of them was difficult with the traditional methods. These frames enabled to create additional coordinate systems to the cleaning process. Thus, the programs can be divided into the smaller parts and the movements could be performed in different coordinate systems in different parts of the program. These additional frames were not just parameterized points as offsets from the base coordinate system, they were also oriented according to the cleaning surface.

All cleaning points were converted to the position variables that were parametrized to these additional frames. Due to the drawbacks of the relative motion commands, also the actual cleaning movements were converted to the position variables with respect to the frames. These changes allowed for many other benefits. First, due to the parameters of the position variables can be modified anywhere in the program, it is no longer necessary to search for a specific motion command to adjust the cleaning. Thus, a total of 30 control parameters were created. Second, because the cleaning movements were determined with respect to the additional frame that was oriented based on the cleaning surface, the orientation of the tool relative to the cleaning surface was also optimized. Thus, the cleaning efficiency of the tool was increased.

In addition to the parametrization of the programs and optimization of the base coordinate system, the commissioning was also expedited with a new rotation feature. As was expressed earlier, the commissioning of the middle spout required a lot of time due to the limit of the A5 axis and the location of the smelt spout relative to the robot. Therefore, a new rotation feature was created where the tool can be rotated for the left wall cleaning of the hood. This gives considerably more clearance to the A5 axis. The possibility of rotation was also added for the right wall cleaning and for the spout cleaning that the cleaning process can be optimized in the different situations. The main changes between the old and new cleaning programs are summarized in the Table 18.

Table 18. Comparison of the old and new cleaning programs.

Cleaning programs	**Old**	**New**
Number or points that should be taught with a real robot	10	0
Relative motion commands	60	2
Frames	0	6
Position variables	0	24
Control parameters	0	30
Tool orientation optimized relative to the cleaning surface	No	Yes
Possibility to use two different orientations of the tool coordinate system in the middle smelt spout	No	Yes

As can be seen from the Table 18, the structure of the cleaning programs changed radically. The accuracy and the success of the development process is evaluated in the following chapter. The decline in the level of competence and the acceleration of the commissioning are analyzed in more detail in the Chapter 4.3.

4.2 Accuracy of the results

The accuracy of the new cleaning programs depends mainly on the accuracy of the base coordinate system. If the location and orientation of the base coordinate system can be determined as accurately as in the virtual world, the cleaning programs should always work perfectly due to parametrization. However, there are always mechanical differences between the real world and the virtual world due to the manufacturing tolerances. Therefore, the control parameters were created.

In any case, the accuracy of the base coordinate system has the biggest impact on the cleaning programs. Therefore, the results of the teaching processes and the success of the cleaning programs should be compared to evaluate the functionality of the whole development process in practice. Based on the values in the Table 13, the difference of the translational coordinates of the base coordinate systems between the teaching processes are calculated on the Tables 19 – 21. The main mechanical differences between the tests are also expressed there. Some of the base coordinate systems were determined with an indirect tool in the second teaching process. Differences that may have been affected by the twisted tool are marked with the symbol “ * “.

Table 19. Differences between the translational coordinates of the base coordinate systems between the first and second teaching process.

Compared teaching processes	**Spout**	**Difference in the translational coordinates**			**To consider**
		X (mm)	**Y (mm)**	**Z (mm)**	
1 → 2	1	-0,61	-0,10	-2,48	Robot was changed
	2*	-0,51	-2,08	-2,63	
	3*	3,61	2,53	1,72	

Table 20. Differences between the translational coordinates of the base coordinate systems between the first and third teaching process.

Compared teaching processes	**Spout**	**Difference in the translational coordinates**			**To consider**
		X (mm)	**Y (mm)**	**Z (mm)**	
1 → 3	1	-3,35	4,20	-9,10	Robot and tool were changed.
	2	-2,95	2,80	-5,95	
	3	5,26	-2,94	6,84	

Table 21. Differences between the translational coordinates of the base coordinate systems between the second and third teaching process.

Compared teaching processes	**Spout**	**Difference in the translational coordinates**			**To consider**
		X (mm)	**Y (mm)**	**Z (mm)**	
2 → 3	1	-2,74	4,30	-6,62	Tool was changed.
	2*	-2,44	4,88	-3,32	
	3*	1,65	-5,49	5,12	

As could be noticed from the tables above, there was variation in the differences. The maximum difference between the components of the origin was 9,1 mm and the minimum was 0,1 mm. Because the measurements results depended on many factors, the exact causes of the differences are difficult to determine. However, it is important to assess the possible causes to improve the accuracy in the future. By assuming that the spout and hood have remained in place between the tests, the possible causes for the difference were the change in the robot's location, the straightness of the tool, human error, the inaccuracy of the teaching methods and manufacturing errors of the hood.

The base coordinate systems were taught at almost to the same point between the first and second teaching process (Table 19). When comparing these processes with the third teaching process (Table 20 & 21), the differences were larger. Due to the exchange of the robot between the first and second tests, it was possible that the direction of the robot changed slightly during the exchange. As was expressed earlier, it has a direct effect on the world coordinate system and thus on the location of the base coordinate systems. However, the maximum difference between the components was 2,48 mm, as was expressed in the Table 19. The base coordinate systems at the second and third spout were not directly comparable because the tool was bent in the second tests. Thus, it seems that the robot exchange had no significant effect on the process. Such a difference may be due to the inaccuracy in the teaching tools (Figure 33) or human error.

It can be assumed that the differences may have been larger at the second and third spout due to the twisted tool (Table 19). The straightness of the tool had a significant effect on the correctness of the points because the main purpose of the new teaching process was to guide the TCP to the desired point visually. Because the TCP of the twisted tool does not correspond to the correct position of the tool coordinate system, the invalid point will be stored to the system, as is shown in the Figure 34 on the left.

The calculation had been made simple because only the translational coordinates of the measured points were collected. In this way, the significance of the approach direction had been removed. On the other hand, if the points are measured with a twisted tool, the approach direction is also important. This could be seen in the Table 19, where the differences were negative at the second spout but positive at the third spout due to the different approach orientation of the tool coordinate system. The situation is visualized in the Figure 34 on the right. For these reasons, it is extremely important to use a direct tool during the teaching processes. Therefore, it is important to teach the tool checkpoint where the straightness of the tool can be checked. However, the recommendation for the future is that the tool should be replaced if it bends.

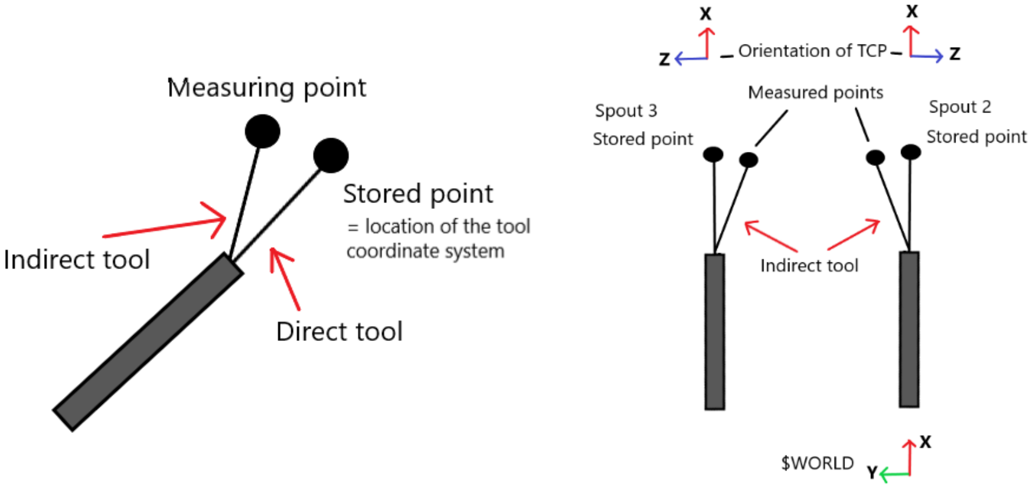


Figure 34. An invalid point will be stored in the system with a twisted tool. The direction of the error depends on the orientation of the tool coordinate system.

The differences increased significantly between the first and third teaching process. According to the Table 20, the differences between the components of the origins were at most 9,1 mm. The differences can be also noticed in the Table 21, where the differences in the Y and Z components of the calculation points are expressed. Also, the range between the differences, the means of the transitions and the maximum deviation from the mean transitions were calculated. The X components were not compared, because there were so dependent on the locations of the measured points.

Table 22. Differences between the translational coordinates of the calculation points between the first and third teaching process.

Spout	Point	Difference between the teaching processes (Process 1 –> Process 3)	
		Difference Y (mm)	**Difference Z (mm)**
1	**P_1**	3,15	-8,12
	P_2	2,98	-6,51
	P_3	3,07	-7,45
Range (mm)		**0,17**	**1,61**
2	**P_1**	3,76	-4,20
	P_2	3,34	-5,37
	P_3	2,67	-5,19
Range (mm)		**1,09**	**1,17**
3	**P_1**	-4,65	3,67
	P_2	-4,94	2,62
	P_3	-3,61	2,02
Range (mm)		**1,33**	**1,65**
Mean of the transitions		**3,57**	**5,02**
Maximum deviation from the mean		**1,37**	**3,10**

As can be noticed from the Table 22, the transitions were large in some components of the points. The maximum displacement was 8,12 mm, which cannot only be explained by the inaccuracy of the measurement tools or human error. The means of the transitions were also high which indicates a mechanical difference between the tests. As was expressed earlier, the new tool was exchanged for the last tests, so perhaps the exchange has affected the differences. In addition, the components have been changed approximately by the same amount (range) at each spout, suggesting that the points have been affected by some external factor. The maximum deviations from the mean values were also so low which can be caused by the inaccuracy of the teaching tool.

Because the new tool can be assumed to be straight, it can be stated that the old tool was already twisted before the first practical test. In reality, the straightness may also vary due to manufacturing tolerances. In other words, it is also possible that a measurement error also occurs with the new tool. The tool can cause errors in the parameters of the points, in which case there is no guarantee that the calculation will be successful. Therefore, the error caused by the tool should be minimized by approaching the calculation points always from the same direction. Thus, the error has the same direction at every point and the orientations can be calculated successfully. In the practical tests, the approach direction of the tool had to be adjusted due to the length and shape of it (Figure 35). That can be prevented in the future by manufacturing a new shorter and sharper teaching tool.

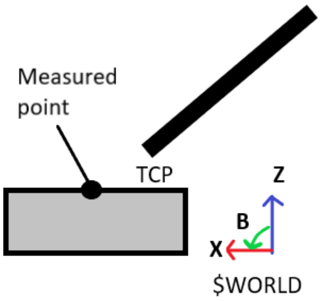


Figure 35. The angle B of the teaching tool should be modified during the teaching process due to the length and shape of the tool.

The accuracy of the measuring points is essential for the entire commissioning because they are used to calculating the orientation of the base coordinate system. In other words, they determine the direction of the movements. The Table 23 summarizes the Euler Angles of the base coordinate systems and expresses the range of the angle differences between the teaching processes. Euler Angles that may have been affected by the twisted tool are marked with the symbol “*“.

Table 23. A summary of Euler Angles and their range between the teaching processes.

Teaching process	**Spout 1 – Euler Angles (degrees)**			**Spout 2 – Euler Angles (degrees)**			**Spout 3 – Euler Angles (degrees)**		
	A	B	C	A	B	C	A	B	C
1	0,85	0,72	-0,17	0,75	-0,62	1,39	0,54	-0,10	1,60
2	0,79	0,51	-0,51	0,78*	-0,55*	1,48*	0,58*	0,11*	1,80*
3	0,83	0,48	-0,40	0,70	-0,46	1,47	0,42	0,06	1,44
Range (degrees)	**0,06**	**0,24**	**0,23**	**0,08**	**0,16**	**0,09**	**0,16**	**0,21**	**0,36**

As was shown in the Table 23, there was variation in the values of the Euler Angles between the teaching processes. The differences were the smallest at the A angles and the largest at the C angles. The angle change of the base coordinate system has a direct effect on the location of the points that are determined with respect to it. The Table 24 provides an example of the effect on a point that is located one meter away from the origin. The differences were calculated using trigonometry.

Table 24. The angle change of the base coordinate system effect on the location of the point that is determined with respect to it. An example point is located one meter away from the origin.

Angle	Minimum angle range (degrees)	Difference at one meter (mm)	Minimum angle range (degrees)	Difference at one meter (mm)
A	0,06	1,05	0,16*	2,79
B	0,16	2,79	0,24	4,19
C	0,09	1,57	0,36*	6,28

According to the Table 24, the minimum angle variation was 0,06 degrees at angle A. It means that the location of the point moves 1,05 mm along Y axis. The maximum angle variation was 0,36 degrees at angle C which means that the point moves 6,28 mm along Z axis one meter away. Differences in the B-angle can also be seen as changes on the Z axis. However, the points are moving along the arc of the circle, so changes also occur on other axes.

The angle changes can be examined also with the smelt spout dimensions such as the length of the spout (730 mm). The teaching accuracy of the angle A can be considered very accurate, because in the commissioning situation, an angle difference of 0,16 degrees would mean 2,04 mm difference between the beginning and the end of the spout. This would be impossible to detect with the human eye. The teaching accuracy of the B angle was also very high because the maximum angle difference between the processes was only 0,24 degrees which mean 3,05 mm difference along the spout. The largest differences were in the measurement of C-angles, presumably due to the short measurement interval. On the other hand, the measurement interval corresponds to the width of the spout. The angle difference of 0,36 degrees can be detected as a 0,78 mm difference in the width of the spout. So even if the angle range of C was larger, the dimensional accuracy was sufficient.

When comparing the differences between the Euler Angles, it is important to understand the influencing factors and look at the consequences of the differences. By considering the twisting of the tool during the second teaching process, the biggest differences in the A and C angles can be ignored (Table 24). Thus, the maximum difference in the A-angles was only 0,10 degrees and at the C-angle 0,24 degrees. Based on the calculation, the values of Euler Angles A and B were calculated from the direction of X axis (Figure 36). Its direction is determined based on the calculation points **P$_2$** and **P$_1$**. Based on the Table 12, the vectors between these points can be formed at different spouts after the first and third teaching processes as is shown in the Table 25.

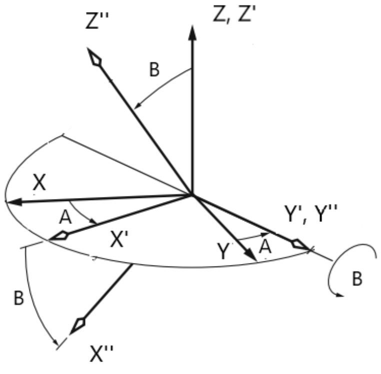


Figure 36. Euler Angles A and B can be calculated based on the direction of the X axis.

Table 25. Vectors between the points $\mathbf{P_2}$ and $\mathbf{P_1}$ after the first and third teaching processes.

Spout	Vector	Teaching process 1	Teaching process 3	Angle difference	
				A (deg)	B (deg)
1	$\mathbf{P_{21}}$	$376{,}78\,\mathbf{i} - 5{,}56\,\mathbf{j} + 4{,}75\,\mathbf{k}$	$372{,}34\,\mathbf{i} - 5{,}39\,\mathbf{j} + 3{,}14\,\mathbf{k}$	-0,02	-0,24
2	$\mathbf{P_{21}}$	$385{,}64\,\mathbf{i} - 5{,}03\,\mathbf{j} - 4{,}20\,\mathbf{k}$	$375{,}36\,\mathbf{i} - 4{,}61\,\mathbf{j} - 3{,}03\,\mathbf{k}$	-0,05	0,16
3	$\mathbf{P_{21}}$	$381{,}74\,\mathbf{i} - 3{,}60\,\mathbf{j} - 0{,}69\,\mathbf{k}$	$368{,}46\,\mathbf{i} - 2{,}72\,\mathbf{j} + 0{,}36\,\mathbf{k}$	-0,12	0,16

Regardless of the mechanical differences such as the exchange of the tool, the difference in the Euler Angles A and B were low. It can be also seen from the vector $\mathbf{P_{21}}$ components **j** and **k** that were very similar between these processes. The **i** component was varying due to the inaccurate points for the calculation. Because the direction of the vector depended on the points $\mathbf{P_2}$ and $\mathbf{P_1}$, it was also depended on the straightness of the measured line between them. Because the points were measured from the surface of the hood, the line was dependent on the accuracy of the hood design.

Even if the points $\mathbf{P_1}$ and $\mathbf{P_2}$ were measured from the determined line, there may be an error in the calculated Euler Angles due to the mechanical errors. The Euler Angle A depends on the error on the Y axis and B depend on the error on the Z axis, as is shown in the Figure 37. It visualizes that the determined angles may not correspond to the orientation of the entire body. The magnitude of the error may also depend on the measurement interval of the points, as the exact location of the mechanical error cannot be known. Thus, the exact locations should be determined for the points in the future. In this way, the reproducibility of teaching processes is improved, as the calculated angles can be re-checked and verified by measuring the points again. One possible way to minimize the possibility of the error, is to calculate the Euler Angles from two different lines and compare the values. The mechanical errors can also affect the calculation of the Euler Angle C (Figure 38), which determines the directions of the Y and Z axes for the base coordinate systems.

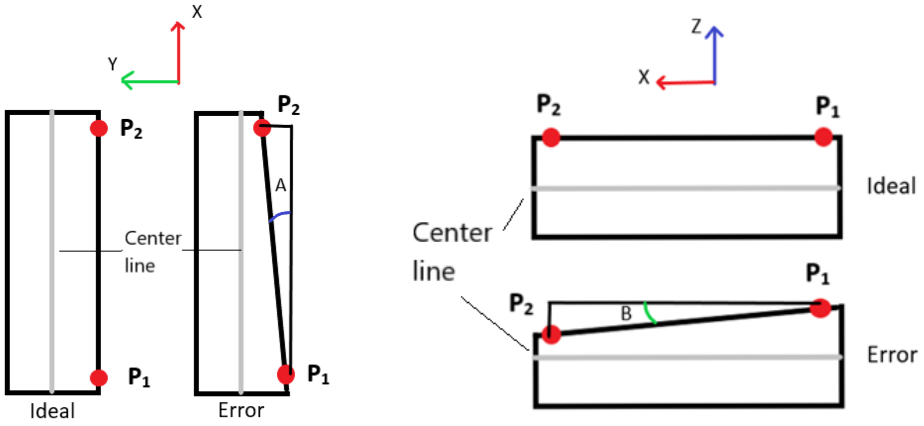


Figure 37. The mechanical errors can affect the values of the Euler Angles.

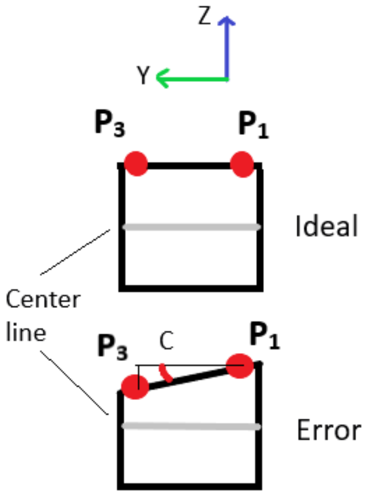


Figure 38. The mechanical error can affect the directions of the Y and Z axes of the base coordinate systems.

The accuracy of the teaching process and the success of the development process can be assessed by looking at the test results of the new cleaning programs. The hood cleaning program was tested only once for scheduling reasons. The program was tested successfully during to the first test. Because the modifications were not needed and the tool moved smoothly along the surfaces, the parametrization of the hood program can be considered successful. However, it should be noted that the hood design differed in test environment from the model that was used during the parametrization. Thus, the cleaning of the hood roof could not be tested. Outside the results of this research, it can be stated that the commissioning of the hood cleaning program has been performed successfully by a robot expert in the test environment of a customer project.

The new parametrized spout cleaning program was tested successfully after each teaching processes. The Table 26 summarizes the modifications from the Table 14 – 16.

Table 26. The modifications in the spout cleaning programs during the practical tests.

Practical test	Spout	Rotations of base coordinate systems changed	Changed Euler Angle	Control parameters changed	Names and directions of changed parameter
1	1	No	-	Yes	Frame of Spout (-Y)
	2	Yes	B	Yes	Frame of Spout (-Y)
	3	No	-	No	-
2	1	No	-	Yes	Frame of Spout (-Y)
	2	Yes	B	No	-
	3	No	-	No	-
3	1	Yes	A	Yes	Frame of Spout (-Y)
	2	Yes	B	Yes	Frame of Spout (-Y)
	3	No	-	No	-

As can be seen in the Table 26, the results correlated with each other between the practical tests even if there were changes in the base coordinate system coordinates (Table 13). The tests showed that there were differences between the hoods and spouts due to the attachment between them. This was observed in each test with the first and second spouts. The mechanical error was corrected by the control parameter that was used the move the cleaning frame of the spout along Y axis. The exception was the second spout cleaning in the second test where was no need to move the frame. This difference can be assumed to be due to tool distortion that correlated the error in the right direction and the error was no longer detected in the human eye.

One critical thing that was noticed in the tests were that the calculated B angle was not suitable for the spout cleaning at the second spout. The tool did not clean the spout at the end as deeply as from the beginning. The same error was repeated in each test but only at the middle spout. Since a similar problem did not occur on the other spouts it can be assumed due to a mechanical difference in the middle spout. Since the hood and spout are fastened to each other with bolts, there should be no difference in orientation between them. In other words, the calculated Euler Angles based on the hood points should also be usable for the spout. Therefore, the error can be caused due to the manufacturing error and thus the angle of the spout did not correspond the parametrized value of the cleaning frame. Other possible reason is the poor structure of the test bench to which the hood and spout were attached. It had not been supported at all in the middle (Figure 32), so the hood may be momentarily bent around the Y axis due to the measuring force. The error was fixed by increasing the magnitude of the calculated B angle.

Even if the modifications were similar between the tests, one additional modification was made in the last test. The calculated Euler Angle A was decreased at the first spout. Although the calculated angle corresponded in value to the first and the second tests (Table 23), the cleaning points were at the end more left than at the beginning. On the other hand, the same error can be assumed to have been affected in the first and second tests because the tool touched the spout surface at the beginning on the left and at the end on the right. Although the spout was cleaned well each time at the first spout, it seems that the calculated A-angles were slightly too large. There are a couple of possible reasons for the difference, a calculation error or the spout was not completely parallel to the hood in the test environment. A calculation error does not seem likely based on the coordinates of the measured points (Table 12). However, it is possible that an error has occurred in the calculation due to external factors such as the straightness of the measured line as was visualized in the Figure 37.

In retrospect, the angle correction method must be considered incorrect, because the angle changes of the base coordinate systems will affect also to the hood cleaning program. There could be unexpected changed in the locations of the hood cleaning frames, if the base coordinate system is adjusted according to the functionality of the spout cleaning. This can lead to difficulties during the commissioning. Therefore, the angle of the spout should be determined by modifying the B angle of the cleaning frame rather than angle of the base coordinate system. Thus, a new control parameter must be created for the future modifications.

From the overall perspective, the development process can be considered successful, as it worked in practically the same way as in the simulation. Although there was a lot of mechanical variation between the tests such as the exchange of the tool and robot, the spout cleaning program worked each time with minor modifications. In addition, because the modifications correlated well with each other, the error were probable due to the mechanical differences. Therefore, the parametrization of the spout cleaning program was successful. Because the functionality of the cleaning programs was dependent on the success of the teaching process and the programs were successful every time, the teaching process can be considered accurate.

4.3 Filling the requirements

When assessing compliance with the requirements of the research, the purposes and the overall research should be considered. The research had three different purposes. One of them was to reduce the time it takes teaching the points in the test environment. The second purpose was to decrease the level of competence required during the commissioning from level 1 to level 2 (Chapter 1.1). In practice, these purposes were to be achieved by parameterizing robot programs and reducing the movements that should be taught. These modifications were made utilizing robotic simulation. Therefore, in the general level, the purpose was identifying the opportunities for the robotic simulation and offer new solutions and methods that could be utilized in the future robot solutions.

According to the overall results of the development process, the number of points to be taught decreased in the cleaning programs from ten to zero. On the other hand, the number of points required to teach the base coordinate system increased from one to four due to the orientation calculation points. In other words, a total of 12 points needs to be taught for the base coordinate systems at the default case in the future. Therefore, the number of points to be taught decreased by only one as is expressed in the Table 27. As the number of the smelt spout increases, the number of points to be taught will be higher in the future than it would have been before.

Table 27. A comparison of the number of the points that should be taught during the commissioning in the past and in the future.

Part of the commissioning	Past	Future
Teaching process	3	12
Cleaning programs	10	0
Total	**13**	**12**

However, it is important to pay attention to the significance of these points and the time required to teach them than the number of them. Previously, the points were located inside the hood, and their teaching required understanding about the performance of the program. Due to the challenging location of the points, their teaching was also time-consuming. The commissioner also needed to know exactly the effect of the modifications. In practice, the points were taught by manipulating the tool for the point that seemed good. Then the success of the point had to be verified by executing the specific cleaning part and evaluate the performance.

In the future, there will not anymore be the points that should be taught inside the hood. The points are located outside the hood very close to each other. They are also easy to teach because the only purpose is to guide the TCP visually to the points on the surface of the hood by following the detailed instructions. Therefore, it can be estimated that the time spent teaching the points has been decreased significantly although the number of points to be taught did not decrease much.

The practical tests showed that the manipulation of the robot can be executed even from the PLC touch screen in the teaching situation. This decreases the level of competence, as the robot commissioner should only know the directions of the world coordinate system. Thus, the robot can be manipulated to the desired point by updating the parameters and pressing the move buttons. According to the results, the moving parameters were not sufficient. The angle of the tool had to be modified due to the shape of the tool. The diameter of the tool head was too large, to direct the TCP to the measurement points from any direction. The optimal approach direction could not be set at the approach point due to the length of the tool and environmental constraints. Therefore, the magnitude of the angle B had to be increased using the robot teach pendant. By manufacturing the shorter teaching tool that is also sharper, these unnecessary angle changes can be avoided. In any case, it can be assumed that the level of competence has dropped to the level 2 (Chapter 1.1) in the base coordinate system teaching process. The Table 27 summarizes the factors that affected the level of competence in the teaching process.

Table 27. Improvements and disadvantages of the new base coordinate system teaching process that affected the required level of competence.

Improvements	**Disadvantages**
• Exact point for the origin. • Orientation is calculated. • Approach program to the spouts. • Robot can be manipulated to the points by modifying parameters. • More accurate teaching process.	• The number of points to be taught increased.

The level of competence required to commission the cleaning programs decreased significantly because it is no longer necessary to manipulate the robot during the commissioning of them. If the base coordinate system is taught successfully and there are no mechanical errors, the cleaning programs should work directly as was observed in this research. However, there are often mechanical differences which is why the cleaning process should be adjusted. Therefore, the control parameters were created. Due to the control parameters, the commissioner should only understand the positive and negative direction of the parameter changes.

The decline in the level of competence is hampered by the number of these parameters. The commissioner should understand the different phases of the cleaning programs that the parameters can be modified successfully. Anyway, it can be assumed that with clear instructions and a short training course, the person from the competence level two (Chapter 1.1) will be able to implement the cleaning programs. The Table 28 expresses the things that affected the level of competence in the new cleaning programs compared with the old ones.

Table 28. Improvements and disadvantages of the new cleaning programs that affected the required level of competence.

Improvements	Disadvantages
• No longer need to use the teach-pendant and manipulate the robot. • No longer need to find specific command from the program to adjust the cleaning. • No longer need to understand the relationships of the parameters. • Cleaning process can be adjusted by the control parameters. • The parameters are descriptively named. • Mechanical difference between the spouts can be easily correlated. • The cleaning process is divided into small parts.	• A lot of control parameters due to the number of different parts of the cleaning programs. • The codes of the programs are long due to the mathematical equations.

When evaluating the success of the research, the whole development process should be considered. Previously, the point teaching required one day in the test environment as was expressed in the beginning. It is important to note that this time was spent teaching the points of all the programs. However, the commissioning of the cleaning programs was the most difficult and time-consuming part. Based on the above arguments, it can be stated that the level of competence that is required during the commissioning of the cleaning programs was dropped to the level 2 (Chapter 1.1).

In addition to changes in the level of competence, efforts were made to accelerate the commissioning. Thus, an unbiased opinion was asked from the robot expert who is the main commissioner of the Smelt Spout Robot. He tested the programs and the new base coordinate system teaching method in two different environments and estimated that the time to teaching the points was halved, as is expressed in the Table 29. According to him, the change could have been larger if the teaching tool had been more accurate.

Table 29. Success of the development areas of the research.

Development area of the Smelt Spout Robot commissioning	In the past	In the future
Level of competence required during the commissioning (Chapter 1.1)	1	2
Time that is spent to teach the points of the movements in the test environment	1 Day	½ Day
Time that is spent to teach the points of the movements on site	1 Day	½ Day

In conclusion, the research showed that the simulation can be used to optimize the commissioning of the industrial robot cell. At the same time, it provided a new way of programming for the future development projects. The utilization of the simulation is discussed in more details in the Chapter 4.5. In any case, the research can be considered successful as the objectives were achieved with the target period of five months. The success would have been greater if similar changes had been made to other programs. Due to the schedule reasons, they were left out the review.

The significance of the research for the company was important because the commissioning of the Smelt Spout Robot was expedited and facilitated. The research was part of the larger development process where the purpose is to decrease the required level of competence during the robot commissioning. The research focused on the part where the robot’s movements are taught because it was time-consuming and significantly increased the required level of the competence. More specifically, the research focused on the most difficult area, cleaning programs.

Due to the development steps, the commissioning was significantly accelerated, as well as the required level of expertise was decreased. The results of the research will generate financial savings for the company in the form of salaries. During the commissioning, the robot’s movements teaching can be outsourced to a person from the second level when a robot expert can concentrate on more expertise needed sections such as the PLC programming. However, it should be noted that there may be problems during commissioning, thus the presence of the robot expert is important. In any case, the entire commissioning schedule can be accelerated, and more cost savings will be generated. A quick schedule increases the customer satisfaction which brings positive publicity to the company.

From the scientific view, the research showed that the measurement tools and 3D models in the simulation environment provide the way to parameterize the robot programs. Due to the current industrial revolution, the number of robots will increase. Thus, the tools offered by digitalization must be exploited to replace the slow traditional methods. By sharing the information about new opportunities and their functionality, the number of users can be increased. Therefore, any research in a relevant area can contribute to accelerating the transition to the era of digitalization.

4.5 Benefits of the robotic simulation

The main goal of the research was to develop the commissioning of the Smelt Spout Robot. The goal was approached by using the robotic simulation. Therefore, the research problem was of how the robotic simulation can be utilized to optimize the commissioning of the Smelt Spout Robot. The research problem was approached by looking for the answers to the research questions. Answers were sought, with a help of the scientific articles and the use of robotic simulation software KUKA Sim 4.0. The first research question was formed as:

- How the simulation can be utilized to simplify and parameterize the robot programs?

The research showed that the simulation enables the parametrization of the robot programs by determining the frames and points to the desired points as offsets with respect to the coordinate systems. As Pan et. al (2012, p. 87-94) were expressed, the 3D models can be utilized to determine the locations of these variables accurately. The accurate measuring tools of the virtual world made it possible to determine the exact dimensions of the points, so they do not to be modified later. By understanding the mathematical background of the coordinate systems and the relationship between them, the coordinate systems where the rotations are determined based on the surface of the 3D models can be determined. The coordinate systems can be determined to the desired point with a few mouse clicks allowing the Euler Angles to be updated automatically.

Once the main structure of the programs has been precisely defined in terms of parameters, the program can be simplified by creating the control parameters around the points of the movements. Utilization of the control parameters is possible only due to the structure of the program is formed using fixed dimensions. When the points are not subjected to human error during the commissioning, the mathematical equations can be formed between them. With a help of calculations, the relationships between parameter changes can be considered and updated automatically. In this way, the simplified programs can be created where is no need to manipulate the actual robot anymore. Therefore, the programs can be optimized to the different environments in advance by creating the digital twin of the robot cell, as the Ružarovský (2019, p. 1-6) and Holubek et al. (2014, p. 63) were expressed in their research. The possible differences between the ideal and the real world can be considered by modifying the values of control parameters.

In a virtual world, everything is ideal, and it does not contain mechanical errors. Therefore, when the simulation is utilized in the real-world projects, it is necessary to understand the difference between the simulation environment and the real life. Therefore, the second research question was formed as:

- Which real-world factors should be considered in the simulation when programming a real robot?

When comparing the differences between the real-world and virtual world, the comparison can be started from the computational origin also known as the world coordinate system. In the default case, the robot is located at the origin of its own world coordinate system. In practice, the orientation of the world coordinate system is determined by the orientation of the robot. In the virtual world, the world coordinate system of the robot can always be determined to be straight with respect to the frame of reference of the layout. In the real world, there is no such benchmark. Because the Smelt Spout Robot was attached to the steel support structure, its orientation was depending on the stability and the orientation of the steel support structure.

Therefore, even if the components are located identical between the real life and the simulation with respect to the robot, their calculated locations may be different with respect to the world coordinate system. Thus, the base coordinate systems are used to define the location of the other components. They enable to determine the locations of the workpieces such as the smelt spouts correctly with respect to the world coordinate system. By specifying the location of the relevant workpieces, the possible error in the world coordinate system can be correlated. A error of the world coordinate system can be noticed in the large displacement movements.

The orientation of the workpieces does not necessarily correspond to the ideal world. They can be rotated relative to the world coordinate system. In the simulation world, the orientation of objects can be easily determined with the measurement tools but in real life, the measurement is not so accurate. Therefore, the additional points must be measured with the real robot and use the calculation to determine the actual orientation as was observed in this research.

Because the purpose of the research was to expedite the robot commissioning where the accuracy is important, the third research question was formed as:

- Which are the advantages and limitations of the robotic simulation and how they affect the accuracy?

As Perez et al. (2020) were expressed that the accuracy of the movements of the robot was low in the first simulation software at the beginning of the millennium but not anymore. The research showed that the accuracy has been developed and the robotic simulation can be used these days to create digital twins and simulate the theoretical model of the robot cell as Ružarovský (2019, p. 1-6) and Datteri & Schiaffonati (2019, p. 117) were mentioned. The pre-planned movements in the simulation worked the same way in the real life. The points that were dimensioned according to the 3D models corresponded well to the reality. Some modifications were needed due to the mechanical errors, but the required modifications were minor. The programs created in the simulation can be also work without modifications in practise as was presented in this study. Due to the parametrization was created utilizing the measuring tools of the robotic simulation, they can be considered accurate. In the real world, equally accurate dimensioning is more difficult, thus the measuring tools are a clear advantage of the robotic simulation.

One of the weaknesses of the simulation is it totality that requires understanding and experience on many issues such as robot programming and 3D modeling. Even if have experience from the robot commissioning in practice, moving a robot in a virtual world can seem to challenge. Therefore, the complexity can affect the usability of the simulation. In addition, the learning process can require time which may affect the interest of companies in utilizing simulation. The interest may also be affected by the costs of the simulation. As de Paula Ferreira (2020) was expressed, the salaries of skilled people are so high that the salary combined with the licensing costs of simulation applications, the total cost can be high. Old traditional working methods may then seem sufficient, especially if the benefits of the simulation are not known. If the utilization of the simulation is an option, the young persons from the beginning of their career may be the best option due to the low salaries and the enthusiasm of young people. This may decrease the accuracy of the simulation due to lack of experience.

On the other hand, the simulation environment can be effective and risk-free teaching tool to robotics, as González-García (2019, p. 923-933) was expressed. There, the robot programming can be learned without fear of collisions and damages. In the simulation, the points of the movements can be visualized in which case the new level in programming can be reached. The mathematical background of the points is easier to understand so their mathematical relationships can be utilized. Therefore, as the number of simulation users increases and the traditional methods are replaced, the accuracy of the robotics increases due to the theoreticality of the simulation.

While the risk-free teaching environment increases the accuracy, there are also some drawbacks. Because the robot can be moved by dragging the arrows of tool coordinate systems in the virtual world, the simulation does not teach how to manipulate the real robot nor consideration of axis limits. In the simulation, the robot can be returned to the home position by pressing the reset button but in reality, the robot should be manipulated using the robot’s teach pendant that could be difficult if does not understand the limitations of the axes. In addition, the simulation does not teach how to avoid dangers because they do not exist in the virtual world. Therefore, when the simulation is used as a teaching tool, it would be ideal to use a real robot alongside it, as in this research. This makes it easier to combine the virtual world and real life. On the other hand, it is possible that the manipulation of the robot becomes easier and similar to the virtual world in the future with the digital revolution.

One of the challenges that can impair the accuracy of the simulation is the software bugs. As the research observed, some robot movements did not work as they should, and the new solutions had to be found. In this research, the errors had a big impact on the whole development process because they guided to search for new ways to program the robot. In any case, the debugging of software bugs can be time-consuming and frustrating.

After all, the overall results showed that the robotic simulation is a capable tool for parametrize the robot programs and optimize the robot cell in the virtual world. It provides opportunities to test the suitability of different robots for the modeled environment and ensure their reachability. With a help of the simulation the movements of the robot can be planned in the virtual world and visualize for the other engineer as Scheidegger et al (2019, p. 475) were also mentioned. When the robot movements are seen by multiple people in the design process, it can be ensured that there are no obstacles on the way of the robot. Finally, the collisions can be checked in the different situations using robotic simulation tools. Thus, the simulation can be used to expedite the entire project schedule by indicating the potential causes of the delay.

It is likely that the use of simulation will become more common when supply increases. When the number of user increases, also the amount of customer feedback is increasing. The increase in the number of users will also increase competition between the software developers. With increasing competition and customer feedback, applications are evolving, and their prices are decreasing. However, it should be noted that other simulations options are available, such as VR simulation that may challenge the traditional robotic simulation applications in the future. The Table 30 summarizes the benefits and limitations of the robotic simulation as a SWOT-analysis.

Table 30. SWOT-analysis of robotic simulation.

Internal	**Strenghts** • Measuring tools. • Reachability tests and the robot selection. • Robot cell optimization. • Movements planning in advance. • Program tests and collision detection.	**Weaknesses** • Software bugs. • License costs. • Learning can be time consuming. • Require understanding of many disciplines.
External	**Opportunities** • Parametrization of old robot programs. • Functionality tests without access to a real robot. • Acceleration of the delivery process. • Effective teaching tool. • Technology development due to the growing competition.	**Threats** • Lack of professionals. • Can reduce the number of people who know how to manipulate the right robot. • VR simulations and new technologies.

As can be seen from the SWOT analysis, the robotic simulation has a lot of strengths and opportunities but there are also weaknesses and threats. It is important to note that the KUKA

Sim 4.0 simulation software was used in the research. The program proved to be a potential tool and will be involved in the new robot development projects in the future. The software bugs had been fixed in the new version of the software that became available at the end of this research.

4.5 Validity of the research

The research methods had been chosen successfully because they were mutually supportive. The research problem was of how the robotic simulation can be utilized to optimize the commissioning of the Smelt Spout Robot. Therefore, the research started by reading the scientific articles on the possibilities of the simulation. Because one of the development targets were the robot programs and their parametrization, it was important to understand the used programming language that the modifications would be possible. Thus, the basics of the KUKA Robot Language were examined. It was also important to understand the mathematical background of the motion programming to parametrize the robot programs, thus it was investigated. Therefore, the literature review offered a comprehensive package of the relevant information for the development process.

The selected simulation software provided an opportunity to learn the programming language in the simulation environment. The simulation tool was selected as KUKA Sim 4.0 that provided compatibility not only with the programming language but also with the robot. Once the programming language was in possession, the case development was started by parametrizing the old programs in the simulation environment. According to the theoretical background the new base coordinate system teaching method was developed and tested in the simulation.

Because the purpose of the research was to develop the commissioning, the development process also had to be tested in practice to evaluate the success and accuracy of it. The test environment resembled the environment of the Smelt Spout Robot which was therefore suitable for testing the new programs. Because the development process worked also in practice and the commissioning can be considered improved, the research methods can be considered suitable for the research. They supported each other and guided the research towards the goals. In addition, they offered the solution to the research problem. Thus, the validity of the research can be considered high.

4.6 Suggestions for the future

Based on the results of the research and the result analysis above, there are several opportunities for the future development. From the general point of view, the robotic simulation would be good to be strongly involved in the design process of robot cells. The first simulation tests can already be made after a visit to the customer's site. Based on the measurements on site, the 3D environment can be modeled and used in the simulation. Therefore, the obstacles that radically interfere the robot's movements can be identified in advance. In this way, the area of the robot cell can be optimized already at the design stage. The use of simulation at a later stage of the robot cell projects is also an advantage, as the robot's transition movements can be planned and tested in virtual world. By transferring the robot programs from the simulation environment to the real robot, the unnecessary teaching work can be avoided, and the time required for commissioning is further reduced.

From the case perspective, the commissioning of the Smelt Spout Robot made significant progress. However, there are still areas that can be developed to expedite the commissioning and decrease the required level of competence. First, by manufacturing a new teaching tool which is sharper and shorter than the actual cleaning tool, the teaching process can be significantly improved. Thus, the suitable approach angles can already be determined to the approach points and there is no need to change the angles during the teaching process anymore. Therefore, it is not necessary to create new parameters (Table 8) for the teaching or use the robot teach-pendant anymore. In this way, the tool can be guided to the measurement points directly from the touch screen of the PLC. Thus, the level of competence required for the teaching process can be decreased.

As was previously expressed, because the PLC is located outside the robot's fences, the manipulation of the robot from the touch screen is slow due to the back-and-forth walking between the robot and PLC. One option could be to utilize the benefits of the digitalization and the remote control of the PLC, by creating a mobile application for the teaching situation. Then the commissioner could manipulate the robot next to it by using a smart phone that can be found in almost everyone's pocket. The mobile application could represent the user interface of the PLC by including a digital remote control. It could include four buttons based on the parameters in the Table 8. It could also contain an emergency stop button, and text fields where the parameters for the movements could be written based on the measurements.

From the commissioner safety view, it would be better to be outside of the robot's fences. If the robot is controlled only from the PLC touch screen, the teaching area and the teaching tool should have cameras to guide the tool to the desired points. There is also possibility to utilize the motion and touch sensors by attaching them to the points to be measured. In that case, a condition can be used in which the measured point would only be accepted if the tool was close enough to the point. This would reduce the possibility of human error in the measurements.

One option could be to take advantage of the laser technology. There could be a laser sensor at the end of the teaching tool that would indicate the distance between TCP and the measuring point. By directing the laser beam towards the desired points and creating the calculation, the parameters of the points could be calculated. In this way, the robot can be manipulated from the PLC touch screen with the help of cameras. When the laser would point accurately enough at the desired location, the point could be stored farther, which would also reduce the risk of collisions.

The teaching process is not the only area that could be further developed. The parametrized programs that have been developed in this research can be exploited in the future to optimize the cleaning processes. By testing the different combinations of the cleaning parameters, the options for the cleaning can be created. They can be also used to find the optimal combinations of cleaning parameters that maximize cleaning quality but minimize tool wear.

The possible cleanup options can be created for the different behaviors of the melt flow. For example, if the width of the melt flow or the solidification varies, there may be different cleaning combinations available for the customer. Such as a strong cleaning, standard cleaning, and soft cleaning where the cleaning depth would be adjusted. The options would allow the customer to influence the quality of cleaning, as in robotic vacuum cleaners.

5. Conclusions

The main objective of the research was to optimize the commissioning of the Smelt Spout Robot. The commissioning had to be accelerated and simplified as it was time-consuming and required a high level of competence in robotics. One of the drawbacks was the robot programs that were made with a traditional robot programming. Therefore, the purpose was also decreasing the required level of competence by parametrizing the robot programs. The robotic simulation was chosen as the main research method. It was used in conjunction with a literature review to find answers to the research questions presented at the beginning.

The robotic simulation enables the utilization of 3D-models shapes as a part of robot programming. Additional coordinate systems that are rotated parallel to the surfaces of the 3D-models can be created for the programs. They can be used to simplify the programs by dividing them into smaller parts. The robot’s motion points can be parametrized with respect to these coordinate systems which allows the programs to be adjusted in parts.

In the virtual world, the environment is always ideal because it can be created with respect to the virtual frame of reference. There is no such possibility in the real life because it includes mechanical differences due to the manufacturing tolerances. These errors create inaccuracies between the simulation and real world and thus affect the accuracy of the programs in practice. On the other hand, it is also possible that the programs can work without necessary modifications, as the research showed.

Therefore, the robotic simulation is the effective tool for creating a digital twin of the robot cells. It can be used already at the design stage to optimize the robot’s environment in advance by visualizing the robot’s movements to other engineers. The accuracy of the simulation is so high that it can also be used in the parametrization of the robot programs. With extremely accurate measuring tools, the robot’s programming points can be set to a precisely defined point. When the basic structure of the programs can be created accurately, mathematical equations can be utilized between the parameters. In this way, the parameter changes can be automated, which reduces the possibility of human error. The biggest weakness of the simulation is the scope of the software that requires an understanding about many disciplines which make it difficult to exploit it in the beginning. However, the number of benefits is so great that it is likely that the new virtual tools will replace the old traditional methods in the future.

In this master’s thesis, significant development steps were taken by utilizing the advantages of the simulation. The commissioning was expedited by half and the required level of expertise decreased significantly by parametrizing the robot programs. In addition, the accuracy of the commissioning was improved by creating the rotation calculation for the coordinate systems. Because the research was conducted during the target period and its purposes were achieved, the research can be considered successful.

References

ABB. 2021. The world's most used offline programming tool for robotics - RobotStudio. [ABB webpage]. [Referred 13.10.2021]. Available: https://new.abb.com/products/robotics/robotstudio

ABB. 2002. ABB'S RobotStudio software enables cost savings in robot-programming. In: Industrial Robot. Emerald Group Publishing Limited, 2022. Vol. 29. Iss. 6. Pp. 552. doi: 10.1108/ir.2002.04929fad.001.

Andritz. 2021. Passion for innovative technologies that shape the world. [Andritz webpage]. [Referred 28.10.2021]. Available: https://www.andritz.com/group-en/about-us

Andritz. 2019. A robotic solution for a rough environment - Smelt Spout Cleaning System for recovery boiler [web document]. Smelt spout cleaning system for recovery boiler. Pulp & Paper. [Referred 27.10.2021]. Available: https://www.andritz.com/resource/blob/333904/a1f713e82327ac63b22c285199667124/download-smelt-spout-robot-leaflet-data.pdf

Bajd, T., Mihelj, M. & Munih, M. 2013. Introduction to Robotics. SpringerBriefs in Applied Sciences and Technology. Springer. 83 p. doi: 10.1007/978-94-007-6101-8.

Bajd, T., Mihelj, M., Lenarčič, J., Stanovnik, A. & Munih, M. 2010. Robotics. Intelligent Systems, Control and Automation: Science and Engineering. Springer. 152 p. doi: 10.1007/978-90-481-3776-3.

Bonev, I. 2017. How is Orientation in Space Represented with Euler Angles? [web document]. Mecademic Robotics. Updated 12.07.2017 [Referred 27.10.2021]. Available: https://www.mecademic.com/en/how-is-orientation-in-space-represented-with-euler-angles

Briot, S. & Khalil,W. 2015. Dynamics of Parallel Robots From Rigid Bodies to Flexible Elements. Switzerland: Mechanisms and Machine Science. Springer. 350 p.

Chang, Y., Devine, K. & Klitzing, G. 2021. Using Virtual Reality for Industrial Robot Programming: A Preliminary Study. In: Journal of Engineering Technology. Published in Washington, American society for engineering education, 2021. Vol. 38. Iss. 1. Pp. 34-44.

Datteri, E. & Schiafonati, V. 2019. Robotic Simulations, Simulations of Robots. In: Minds and Machines. Springer, 2019. Vol. 29. Iss. 1. Pp. 109-125. doi: 10.1007/s11023-019-09490-x

Delfoi. 2021. Delfoi Robotics software. [Delfoi webpage]. [Referred 13.10.2021]. Available: https://www.delfoi.com/delfoi-robotics/delfoi-robotics-software/

de Paula Ferreira, W., Armellini, F. & De Santa-Eulalia, L. A. 2020. Simulation in industry 4.0: A state-of-the-art review. In: Computers & Industrial Engineering. Elsevier, 2020. Vol. 149. Article number 106868. doi: 10.1016/j.cie.2020.106868.

Fanuc. 2021. FANUC ROBOGUIDE Simulation Software. ROBOT PRODUCTS. [Fanuc webpage]. [Referred 13.10.2021]. Available: https://www.fanucamerica.com/products/robots/robot-simulation-software-FANUC-ROBOGUIDE

Flores, P. 2015. Concepts and Formulations for Spatial Multibody Dynamics. SpringerBriefs in Applied Sciences and Technology. Springer. 83 p. doi: 10.1007/978-3-319-16190-7.

Gasparetto, A. & Scalera, L. 2018. From the Unimate to the Delta Robot: The Early Decades of Industrial Robotics. In: Explorations in the History and Heritage of Machines and Mechanisms. Proceedings of the 2018 HMM IFToMM Symposium on History of Machines and Mechanisms. Springer, 2019. Vol. 37. Pp. 284-295. doi: 10.1007/978-3-030-03538-9_23

Goel, R. & Gupta, P. 2019. Robotics and Industry 4.0. In: A Roadmap to Industry 4.0: Smart Production, Sharp Business and Sustainable Development. Springer, 2019. Pp. 157-169. Part of the Advances in Science, Technology & Innovation book series.

González-García, S., Rodríguez-Arce, J., Loreto-Gómez, G. & Montaño-Serrano, V. M. 2019. Designing a teaching guide for the use of simulations in undergraduate robotics courses: a pilot study. In: International Journal on Interactive Design and Manufacturing (IJIDeM). Springer, 2019. Vol. 13. Iss. 3. Pp. 923-933. doi: 10.1007/s12008-019-00551-0

Holubek, R., Sobrino, D. R. D., Košťál, P. & Ružarovský, R. 2014. Offline programming of an ABB robot using imported CAD models in the RobotStudio software environment. In: Applied mechanics and materials. Switzerland, Trans Tech Publications, 2014. Vol. 683. Pp. 62-67. doi: 10.4028/www.scientific.net/AMM.693.62

International Federation of Robotics (IFR). 2021. China leads post-pandemic recovery. IFR Press Room. [IFR webpage]. Updated 01.07.2021 [Referred 13.9.2021]. Available: https://ifr.org/ifr-press-releases/news/china-leads-post-pandemic-recovery

IFR. 2021. Robot History. [IFR webpage]. [Referred 13.9.2021]. Available: https://ifr.org/robot-history

IFR. 2019. WR Industrial Robots – Sources & Methods [web document]. [Referred 13.09.2021]. Available: https://ifr.org/downloads/press2018/WR%20Industrial%20Robots%202019_Chapter_1.pdf

ISO 8373. 2012. Robots and robotic devices -Vocabulary. 2nd edition. 38 p.

KUKA. 2021a. The history of KUKA. [KUKA webpage]. [Referred 13.09.2021]. Available: https://www.kuka.com/en-se/company/about-kuka/history

KUKA. 2021b. KR 250 R2700-2 C. [web document]. Updated 18.10.2021. [Referred 26.10.2021]. Available: https://www.kuka.com/-/media/kuka-downloads/imported/6b77eecacfe542d3b736af377562ecaa/0000350217_en.pdf?rev=75f32a23be2f4a6d8968204103fe7a00&hash=376E89443211384F5B03364DFEC795D8

KUKA. 2021c. KUKA Sim 4.0. [KUKA webpage]. [Referred 27.10.2021]. Available: https://www.kuka.com/en-de/products/robot-systems/software/planning-project-engineering-service-safety/kuka_sim

KUKA. 2021d. KUKA OfficeLite. [KUKA webpage]. [Referred 27.10.2021]. Available: https://www.kuka.com/en-de/products/robot-systems/software/planning-project-engineering-service-safety/kuka_officelite

KUKA. 2021e. Operating and Programming Instructions for System Integrators. KUKA System Software 8.6 [web database]. Seventh version. 753 p. Service is chargeable and needs user license.

Pan, Z., Polden, J., Larkin, N., Van Duin, S. & Norrish, J. 2012. Recent progress on programming methods for industrial robots. In: Robotics and Computer-Integrated Manufacturing. Elsevier, 2012. Vol. 28. Iss. 2. Pp. 87-94. doi: 10.1016/j.rcim.2011.08.004

Perez, L., Rodríguez-Jiménez, S., Rodríguez, N & Usamentiaga, R. 2020. Digital Twin and Virtual Reality Based Methodology for Multi-Robot Manufacturing Cell Commissioning. In: Applied sciences. MDPI (Multidisciplinary Digital Publishing Institute), 2020. Vol. 10. Iss. 10. Pp. 3633. doi: 10.3390/app10103633

Perez, L., Diez, E., Usamentiaga, R. & García, D. F. 2019. Industrial robot control and operator training using virtual reality interfaces. In: Computers in Industry. Elsevier, 2019. Vol. 109. Pp. 114-120. doi: 10.1016/j.compind.2019.05.001

RoboDK. 2021. Simulator for industrial robots and offline programming – RoboDK. [RoboDK webpage]. [Referred 26.10.2021]. Available: https://robodk.com/index

Ružarovský, R., Holubek, R., Sobrino, D. R. D. & Velíšek, K. 2019. A case study of robotic simulations using virtual commissioning supported by the use of virtual reality. Proceedings of Modern Technologies in Manufacturing. MATEC Web of Conferences, held at Sibiu, Romania 5-7 June, 2019. Vol. 299. Article Number 02006. doi: 10.1051/matecconf/201929902006

Scheidegger, A. P., Pereira, T. F., de Oliveria, M. L. M., Banerjee, A. & Montevechi, J. A. B. 2018. An introductory guide for hybrid simulation modelers on the primary simulation methods in industrial engineering identified through a systematic review of the literature. In: Computers & Industrial Engineering. Elsevier, 2018. Vol. 124. Pp. 474-492. doi: 10.1016/j.cie.2018.07.046

Shabana, A. 2005. Dynamics of Multibody Systems. Third edition. Cambridge: Cambridge University Press. 374 p.

Tran, H., Sunil, A. & Jones, A.K. 2006. The Fluidity of Recovery Boiler Smelt. In: Journal of Pulp and Paper Science. Review Panel, 25 May, 2006. Vol. 32. Iss. 3. Pp. 182-187.

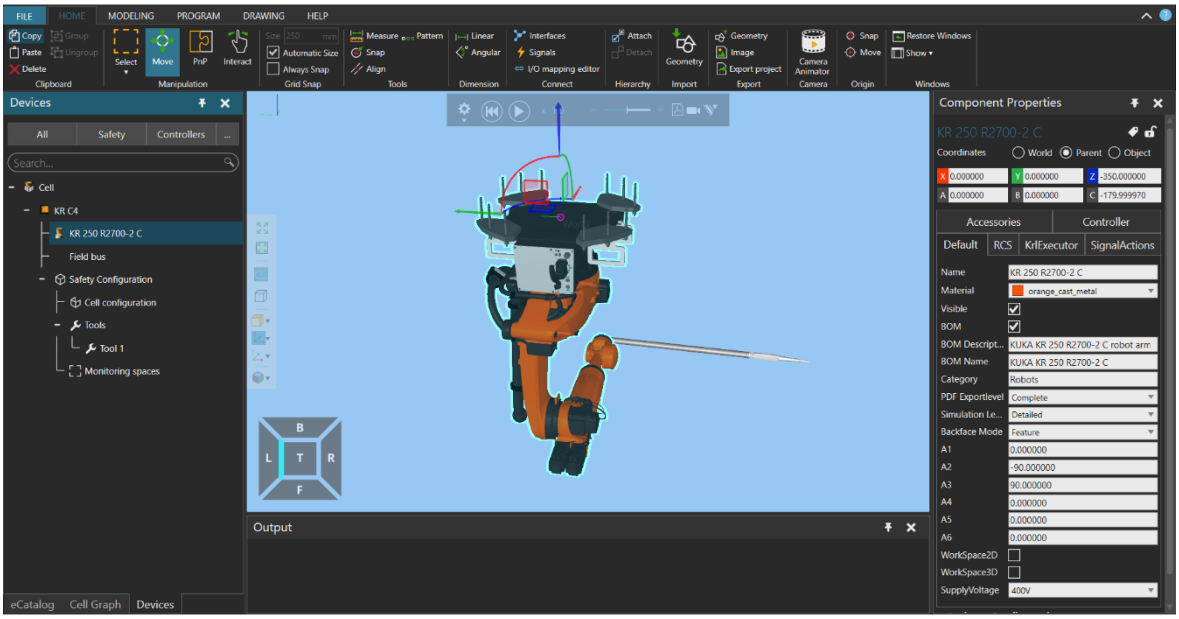
Yaskawa. 2021. MotoSim EG-VRC - Offline programming system with 3D simulation. Robotics Software. [Yaskawa webpage]. [Referred 26.10.2021]. Available: https://www.yaskawa.eu.com/products/software/productdetail/product/motosim-eg-vrc_1686

Vakkilainen, E. K. 2017. Steam generation from biomass: construction and design of large boilers. Amsterdam, Netherlands: Butterworth-Heinemann. 324 p.

Visual Components. 2021. Visual Components 4.3. [Visual Components webpage]. [Referred 27.10.2021]. Available: https://www.visualcomponents.com/products/

W.A.T. Solutions. 2021. Industrial Grade Simulation Software for Education. Workspace – Robot Simulation. [W.A.T. Solutions webpage]. [Referred 13.10.2021]. Available: http://workspacelt.com/

Appendix 1: User interface of KUKA Sim 4.0



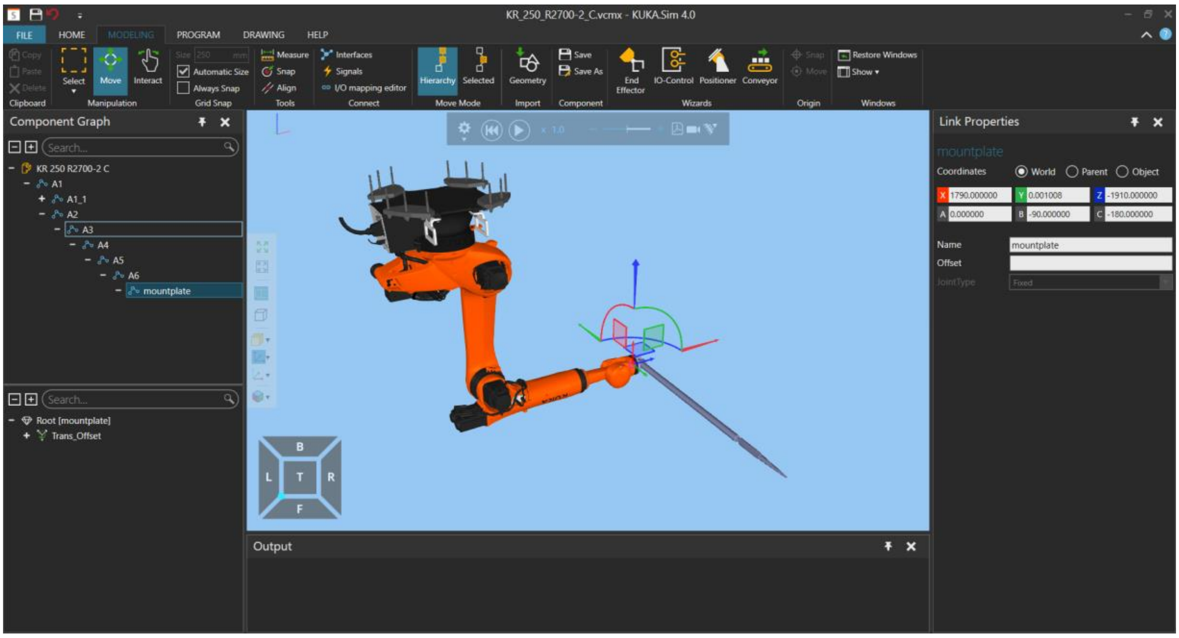
KR 250 R2700-2 C

Coordinates ○ World ● Parent ○ Object

X 0.000000	Y 0.000000	Z -350.000000
A 0.000000	B 0.000000	C -179.999970

Name	KR 250 R2700-2 C
Material	orange_cast_metal
Visible	☑
BOM	☑
BOM Descript...	KUKA KR 250 R2700-2 C robot arm
BOM Name	KUKA KR 250 R2700-2 C
Category	Robots
PDF Exportlevel	Complete
Simulation Le...	Detailed
Backface Mode	Feature
A1	0.000000
A2	-90.000000
A3	90.000000
A4	0.000000
A5	0.000000
A6	0.000000
WorkSpace2D	□
WorkSpace3D	□
SupplyVoltage	400V

Figure 1. Home tab.



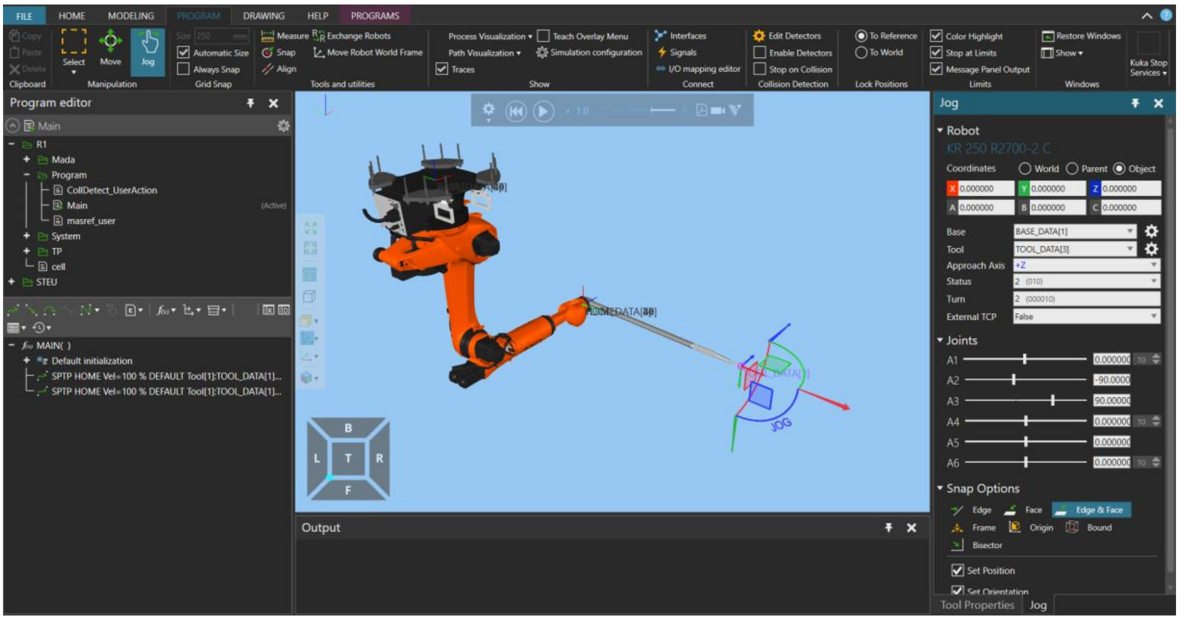
mountplate

Coordinates ● World ○ Parent ○ Object

X 1790.000000	Y 0.001008	Z -1910.000000
A 0.000000	B -90.000000	C -180.000000

Name	mountplate
Offset	
JointType	Fixed

Figure 2. Modeling tab.



Program editor

- Main
- R1
  - Mada
  - Program
    - CollDetect_UserAction
    - Main (Active)
    - masref_user
  - System
  - TP
  - cell
- STEU

MAIN( )
- Default initialization
- SPTP HOME Vel=100 % DEFAULT Tool[1]:TOOL_DATA[1]...
- SPTP HOME Vel=100 % DEFAULT Tool[1]:TOOL_DATA[1]...

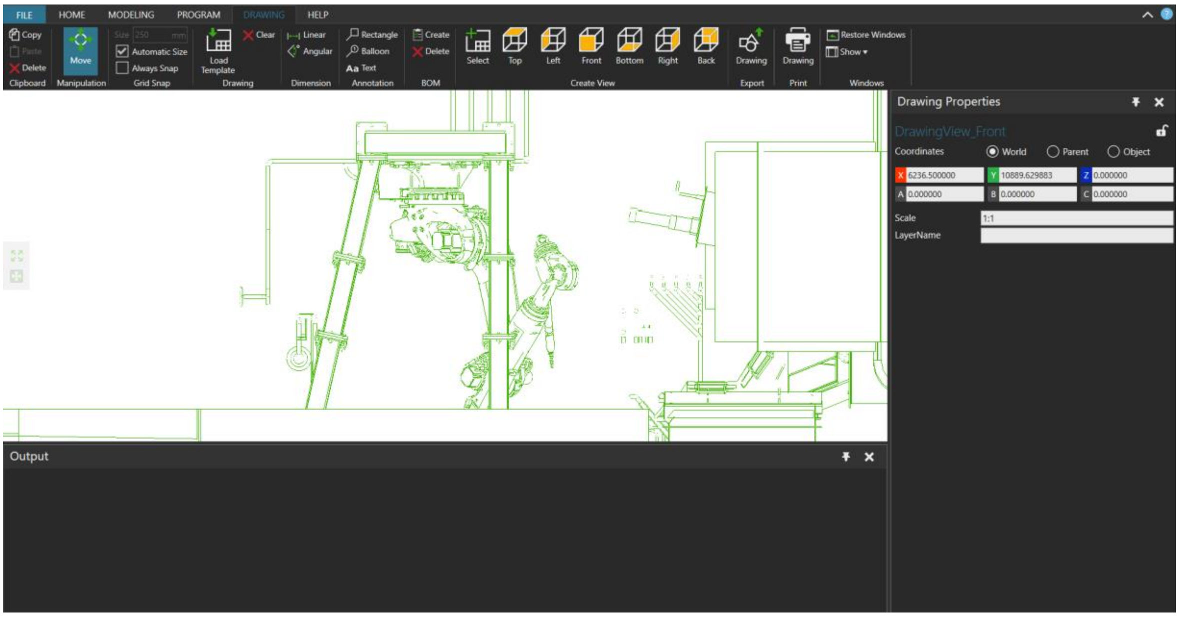
Jog

Robot
KR 250 R2700-2 C
Coordinates ○ World ○ Parent ● Object
X 0.000000 Y 0.000000 Z 0.000000
A 0.000000 B 0.000000 C 0.000000
Base BASE_DATA[1]
Tool TOOL_DATA[3]
Approach Axis +Z
Status 2 (010)
Turn 2 (000010)
External TCP False

Joints
A1 0.000000
A2 -90.0000
A3 90.0000
A4 0.000000
A5 0.000000
A6 0.000000

Snap Options
Edge Face Edge & Face
Frame Origin Bound
Bisector
☑ Set Position
☑ Set Orientation

Figure 3. Program tab.

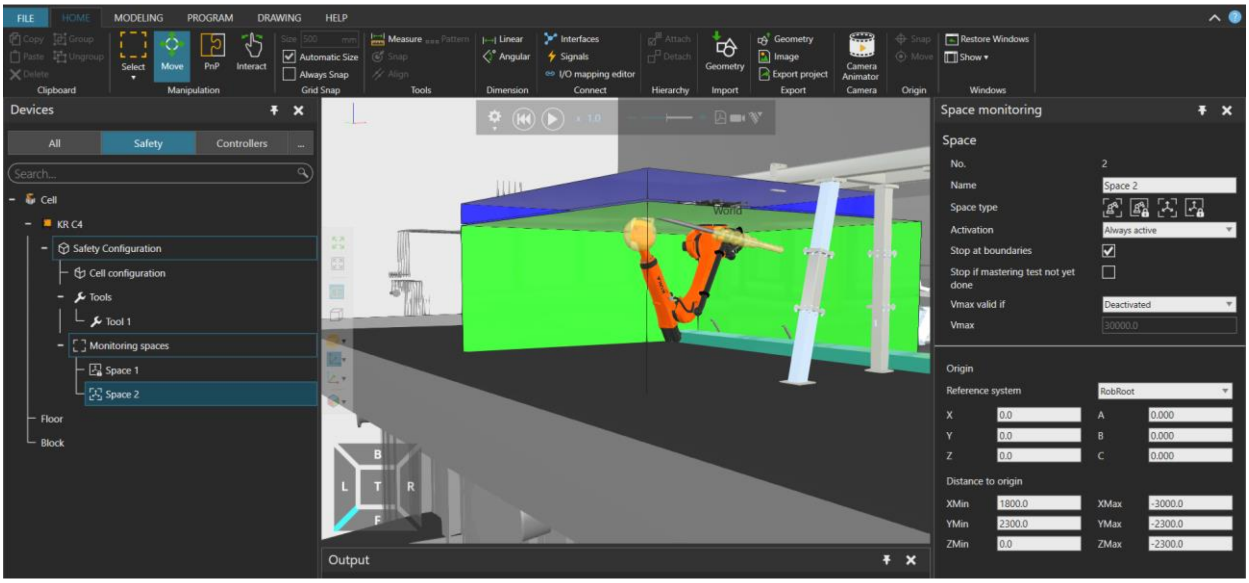


Drawing Properties

DrawingView_Front
Coordinates ● World ○ Parent ○ Object
X 6236.500000 Y 10889.629883 Z 0.000000
A 0.000000 B 0.000000 C 0.000000
Scale 1:1
LayerName

Figure 4. Drawing tab.

Appendix 2: Safety devices in KUKA Sim 4.0.



Space monitoring

Space

No. 2

Name: Space 2

Space type

Activation: Always active

Stop at boundaries ☑

Stop if mastering test not yet done □

Vmax valid if: Deactivated

Vmax: 30000.0

Origin

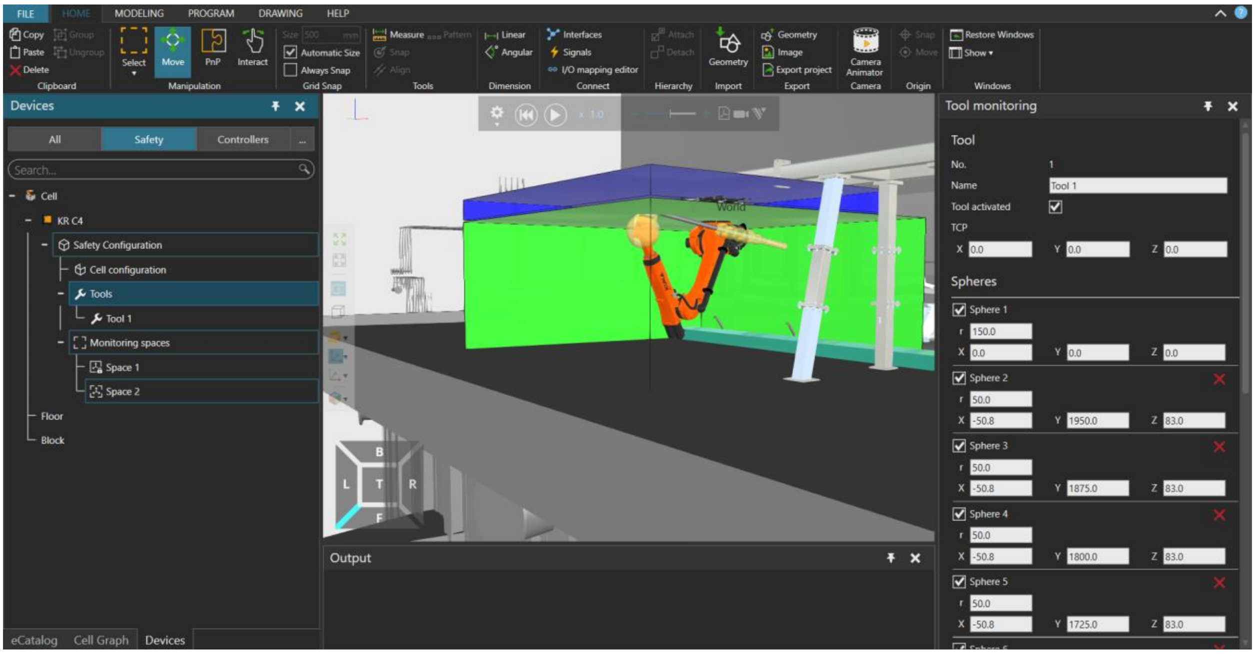
Reference system: RobRoot

X	0.0	A	0.000
Y	0.0	B	0.000
Z	0.0	C	0.000

Distance to origin

XMin	1800.0	XMax	-3000.0
YMin	2300.0	YMax	-2300.0
ZMin	0.0	ZMax	-2300.0

Figure 1. Safety Devices: Space monitoring.



Tool monitoring

Tool

No. 1

Name: Tool 1

Tool activated ☑

TCP

X 0.0 Y 0.0 Z 0.0

Spheres

☑ Sphere 1
r 150.0
X 0.0 Y 0.0 Z 0.0

☑ Sphere 2
r 50.0
X -50.8 Y 1950.0 Z 83.0

☑ Sphere 3
r 50.0
X -50.8 Y 1875.0 Z 83.0

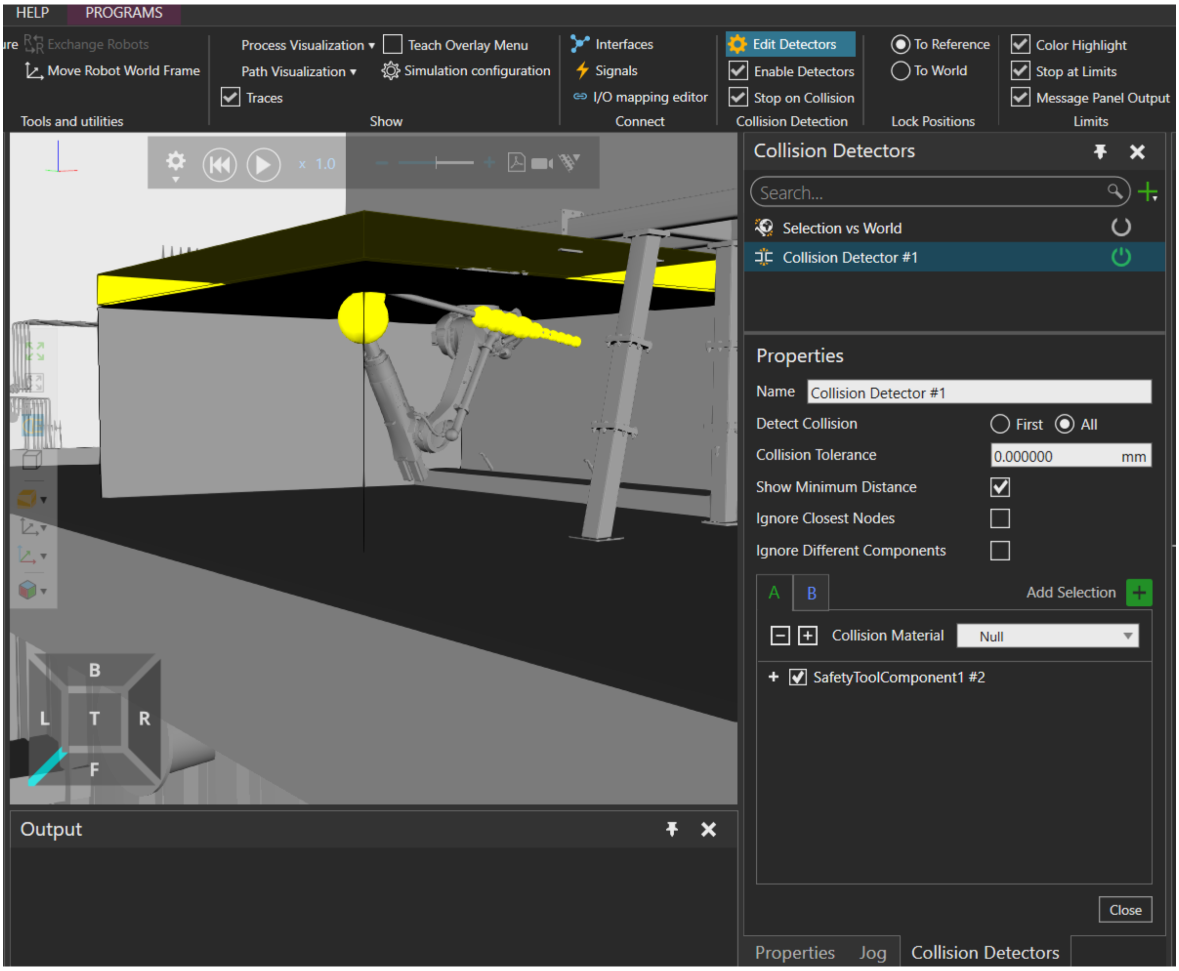
☑ Sphere 4
r 50.0
X -50.8 Y 1800.0 Z 83.0

☑ Sphere 5
r 50.0
X -50.8 Y 1725.0 Z 83.0

☑ Sphere 6

Figure 2. Safety Devices: Tool monitoring.

Appendix 3: Collision detection in KUKA Sim 4.0.



□ Teach Overlay Menu

☑ Traces

☑ Enable Detectors

☑ Stop on Collision

● To Reference

○ To World

☑ Color Highlight

☑ Stop at Limits

☑ Message Panel Output

Name: Collision Detector #1

Detect Collision: ○ First ● All

Collision Tolerance: 0.000000 mm

Show Minimum Distance ☑

Ignore Closest Nodes □

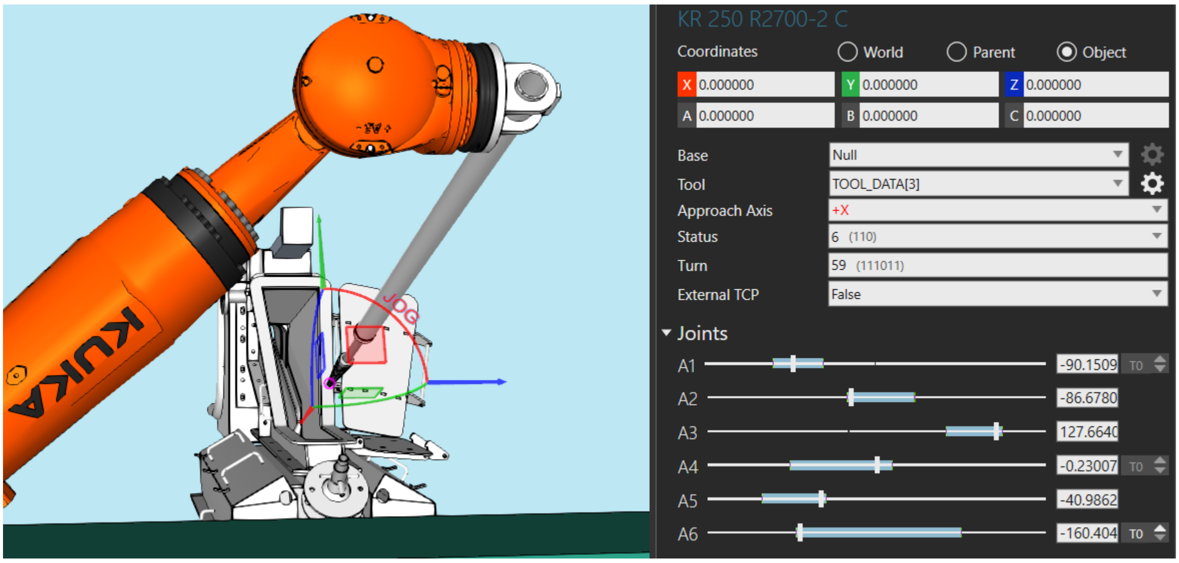
Ignore Different Components □

Collision Material: Null

+ ☑ SafetyToolComponent1 #2

Figure 1. A collision is highlighted in yellow.

Appendix 4: Orientation of the tool coordinate system.



KR 250 R2700-2 C

Coordinates ○ World ○ Parent ● Object

X 0.000000 Y 0.000000 Z 0.000000

A 0.000000 B 0.000000 C 0.000000

Base Null

Tool TOOL_DATA[3]

Approach Axis +X

Status 6 (110)

Turn 59 (111011)

External TCP False

Joints

A1 -90.1509 T0

A2 -86.6780

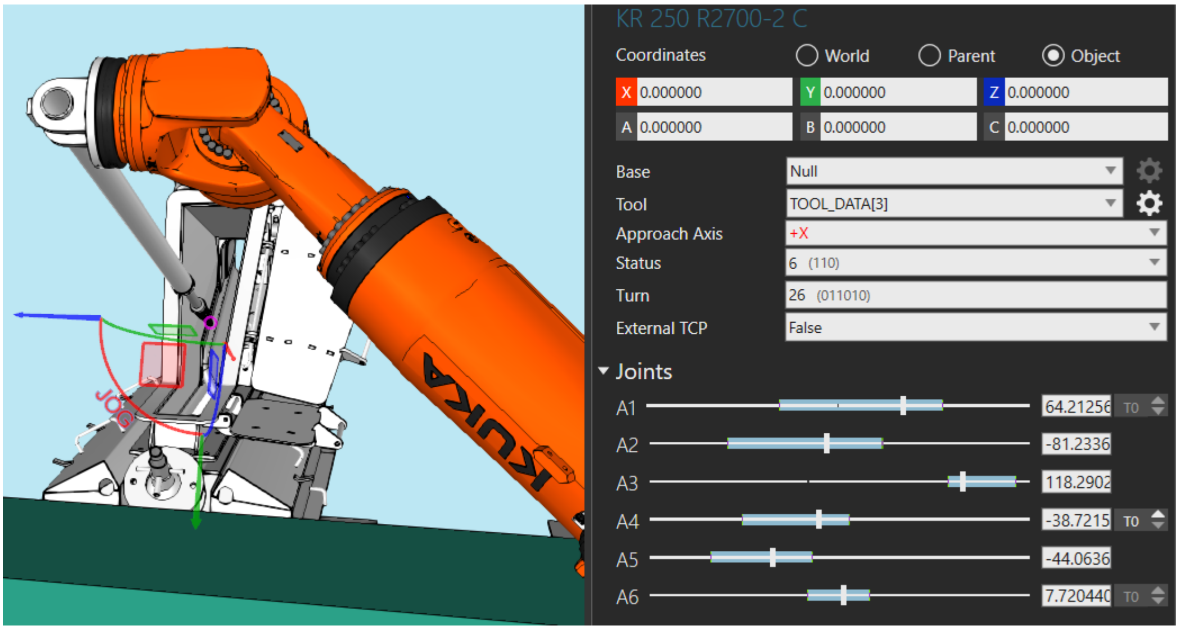
A3 127.6640

A4 -0.23007 T0

A5 -40.9862

A6 -160.404 T0

Figure 1. Tool coordinate system orientation at the first spout.



KR 250 R2700-2 C

Coordinates ○ World ○ Parent ● Object

X 0.000000 Y 0.000000 Z 0.000000

A 0.000000 B 0.000000 C 0.000000

Base Null

Tool TOOL_DATA[3]

Approach Axis +X

Status 6 (110)

Turn 26 (011010)

External TCP False

Joints

A1 64.21256 T0

A2 -81.2336

A3 118.2902

A4 -38.7215 T0

A5 -44.0636

A6 7.720440 T0

Figure 2. Tool coordinate system orientation at the third spout.