FCTUC FACULDADE DE CIÊNCIAS
E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

DEPARTAMENTO DE
ENGENHARIA MECÂNICA

Design and calibration of a multi-robot work cell

Submitted in Partial Fulfilment of the Requirements for the Degree of Master in Mechanical Engineering in the speciality of Production and Project

Author
Gil Boyé de Sousa

Supervisors
Mr. Adel Olabi, Professor from Arts et Métiers ParisTech – Lille Campus
Mrs. Ana Paula Bettencourt Martins Amaro, Professor from Universidade de Coimbra

Jury

President	**Mr. José Luís Ferreira Afonso** **Professor from Universidade de Coimbra**
Examiner	**Mr. Pedro Mariano Simões Neto** **Professor from Universidade de Coimbra**
Supervisor	**Mrs. Ana Paula Bettencourt Martins Amaro** **Professor from Universidade de Coimbra**

Institutional Collaboration

Arts et Métiers ParisTech – Lille Campus

Coimbra, September, 2015

To my parents, brother, sister and girlfriend.

Aos meus pais, irmão, irmã e namorada.

ACKNOWLEDGEMENTS

First, I would like to express my deepest gratitude to my advisor, Mr. Adel Olabi, for his excellent guidance, caring, patience, and providing me an excellent atmosphere for doing this work. He was always ready to give me the technical support needed and became more of a mentor and friend, than simply an advisor.

I am also grateful to Mrs. Ana Amaro that was always ready to answer me the questions and doubts that I had along this master thesis. I must also thank Mr. Pedro Neto and Mr. José Luís Afonso for being part of my thesis jury and for their kindness along this master thesis.

A very special thanks goes out to Jorge, Adrien and Jean Paul, my “colleagues” at the laboratory since the beginning of my internship, with whom I exchanged many ideas and learned a lot during all the time spent in the laboratory.

I must also thank Mr. Olivier Gibaru and Mr. Eric Nyiri for sharing all their knowledge, each one on its expertise. Eric was very helpful and transmitted me many of his immense knowledge in Programming. I keep good memories of the calculation of the cylindrical equation in the 3D space with Olivier.

I also want to thank Mr. Richard Béarée for his kindness and for the moments of conviviality during the lunches.

In addition to this, I must thank Claude who were very helpful to all my needs concerning machining and electrical connections.

I am also grateful to my housemates and friends that were always very encouraging and an important support while living in Lille.

I would like to thank my family: my parents and my sister for supporting me throughout this master thesis and my life in general.

Last but not the least, I would like to thank my girlfriend for its support, encouragement and patience during all my work.

CONFIDENTIALITY STATEMENT

This work is based on confidentiality, containing internal data belonging to an enterprise and so it cannot be used by third parties without the author or the enterprise consent. In Chapters 3 and 4, some of the images and information are hidden due to confidentiality.

Abstract

The main objective of the work presented is the design of a robotic cell and its calibration. This robotic cell needs to be compact, flexible and collaborative. Concerning these constraints, an optimization work was performed using a numerical simulation software applied to industrial robots (V-REP). In addition to this, a calibration model was developed for this robotic cell using a single point laser measurement sensor. Two different methods were used to the calibration: one applied to define the position and orientation of the work-objects in the robotic cell and the other to determine the position and orientation of a cylindrical form.

Keywords: Robotic cell design, Robotic cell calibration, Co-manipulation, Collaborative robotics, Single point laser measurement sensor, V-REP.

Resumo

O objectivo deste trabalho consiste na concepção de uma célula robotizada e na sua calibração. Esta célula necessita ser compacta, flexível e colaborativa. Perante estas restrições foi realizado um trabalho de optimização utilizando um software de simulação numérica aplicado a robôs industriais (V-REP). Além disso, um modelo de calibração foi desenvolvido para esta célula robotizada utilizando um sensor laser de medida. Dois métodos diferentes foram usados para a calibração: um deles aplicado para definir a posição e orientação dos objectos na célula robotizada e outro para determinar a posição e orientação de uma forma cilíndrica.

Palavras-chave: Concepção de Célula Robotizada, Calibração de Célula Robotizada, Co-manipulação, Robótica Colaborativa, Sensor Laser de Medida, V-REP.

Contents

LIST OF FIGURES

1. INTRODUCTION

Industrial robots can be seen in any kind of industry performing any type of task like painting, welding or pick-and-place operations. It is very useful within a productive system and it makes companies earn more money and work with good efficiency. In Figure 1.1 is presented an example of an industrial robot which is able to perform repetitive and painful tasks. In addition to this, it can work under risky conditions like high temperature or non-breathable environments. Those are the main advantages of a robot, but there is one important drawback which is its lack of accuracy. For some specific tasks, it is demanded to have a good precision and we must ensure that the industrial robot performs the desired task with the best possible accuracy.

Figure 1.1 Industrial Robot

Repeatability and **accuracy** are both terms that characterize the positioning capability of a robot (see Figure 1.2).

Repeatability is the capability to return to a defined pose several times and each time this pose is reached the difference in comparison to the other times is minimal.

Accuracy stands for the difference between a nominal pose desired ordered by the controller and the actual pose obtained by the robot [1].

Looking to Figure 1.2, the industrial robots work like shown in the example with good repeatability and poor accuracy. Its repeatability ranges vary between 0.03 and 0.2 mm on average, while the robot can commit errors of several millimeters concerning the accuracy. There is a set of factors contributing to this issue: geometric and nongeometric factors. The geometric ones consider the geometric parameters, the joints offset and the Tool Centre Point definition. The origin of these deviations is the manufacturing process of the robot, which means that the real geometry of the robot components does not match the previously designed and projected geometry. Supposing that each robot link is 1 mm bigger than it was projected, the error of the end-effector pose will be an accumulation of all this errors. In addition to this, when the robot is assembled, the gearboxes and transmissions can produce errors due to its backlashes and its manufacturing errors too. The nongeometric errors are mostly dependent of the robot configuration. These errors are caused by the compliance of the robot links, gearboxes backlashes, kinematic errors, encoder resolution and thermal effects [2]. To sum up, calibration is a fundamental procedure to improve the performance of an industrial robot.

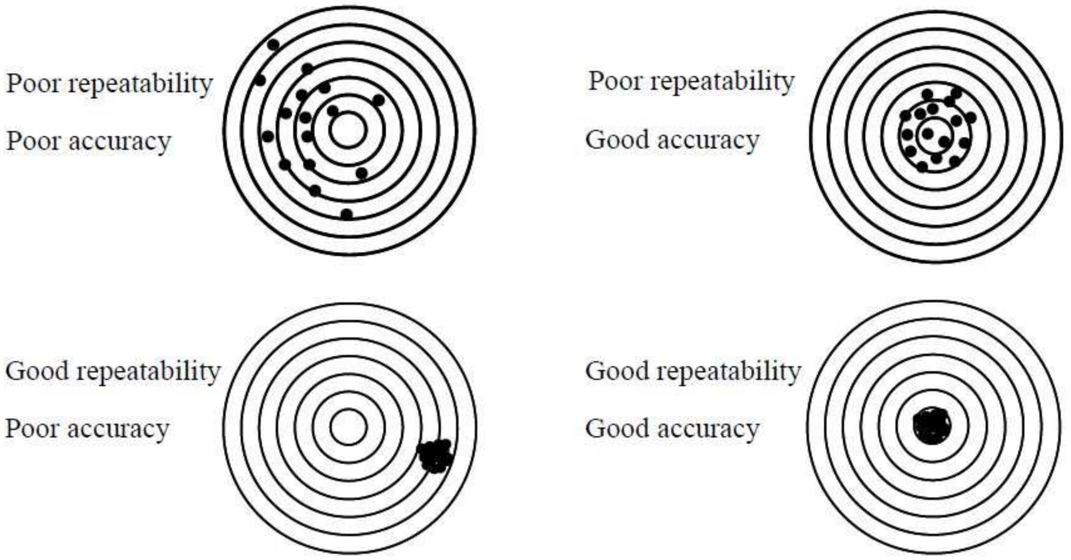


Figure 1.2 Repeatability and accuracy

1.1. Geometric models

The desired locations of a robot end-effector are normally specified in Cartesian space, while these locations are achieved by controlling the joint variables in the robot's joint space. A geometric model is needed to relate the joint displacements with the pose of the end-effector. The absolute accuracy of the robot depends on how accurately this model represents the actual robot. For a given set of joint coordinates, the **direct geometric model (DGM)** consists of solving it in order to obtain the corresponding set of end-effector coordinates. On the other hand, the **inverse geometric model (IGM)**, for a given set of end-effector coordinates, it gives the corresponding joint coordinates. Figure 1.3 illustrates how the geometric models work for a 6 Degrees of Freedom robot. One important difference between these models is that the DGM generates only one solution, while the IGM can obtain more than one solution, as in some cases different robot configurations can reach the exact same end-effector pose [3]. This only occurs inside the robot workspace, while outside the robot workspace there is no solution for the IGM.

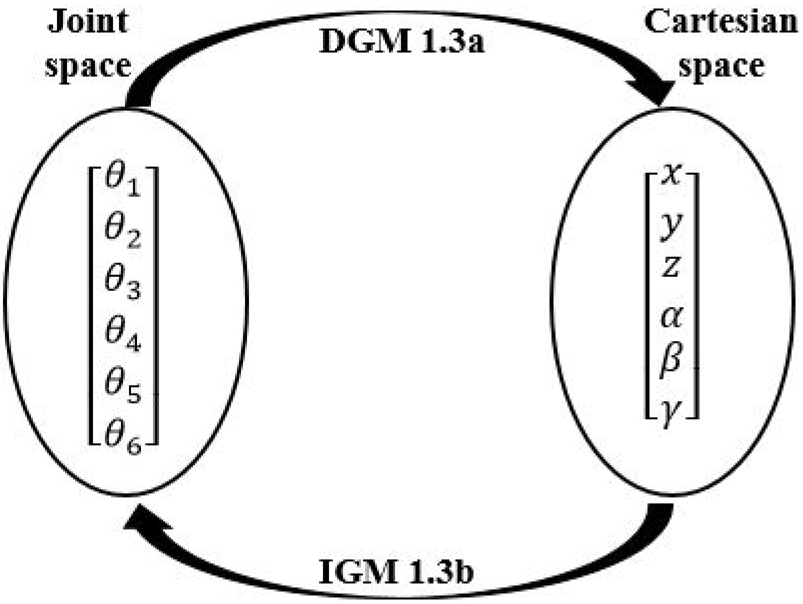


Figure 1.3 Geometric models (6 DOF robot)

1.2. Robot calibration definition

Robot calibration is a procedure in order to improve the accuracy of positioning reducing the difference between the theoretical pose and the real one [4] [5].

In general, there are 4 steps to calibrate a robot:

- Modelling: consists on the elaboration of a model that allows to establish a mathematical relationship between the joint variables and the resulting pose of the end-effector. It must contain the relevant factors that will increase the accuracy;

- Measurement: this step is the collection of data from measurements of the robot. It gives the real poses of the end-effector which can then be compared to the theoretical poses to evaluate the inaccuracy level of the end-effector poses;
- Identification: it is necessary to select the parameters to modify in order to improve the accuracy of the end-effector pose using numerical methods;
- Compensation: it is the application of the result of the last 3 steps directly on the controller of the robot to reduce the difference between the desired pose and the actual pose.

1.3. Robotic cell and offline programming definitions

A robotic cell is a system composed by robots, tools, work pieces, conveyors or structures dedicated to a specific task. It can be used to perform any kind of tasks on the industry (see Figure 1.4). A robotic cell usually is designed with the help of an offline programming software. Offline programming consists in simulating with a computer the feasibility of the tasks desired. This allows to get an idea of the behavior of the robotic cell before implementing it. When using offline programming it is critical to have an efficient calibration procedure before using the robotic cell in the real world.

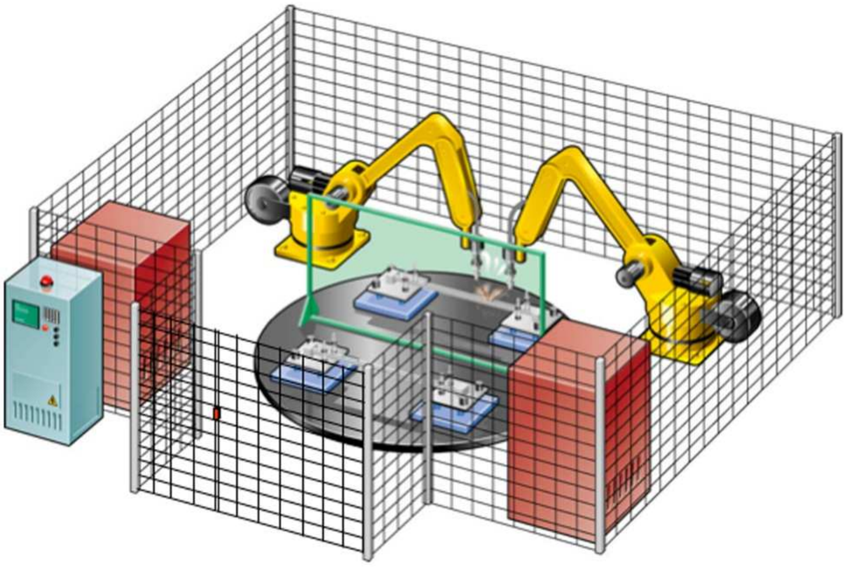


Figure 1.4 Example of a Robotic cell

1.4. Positioning of the work

After getting in touch with the Robotics field it is time to explain what the contributions of this work were. The work presented on this report is divided in three main parts and its conclusions.

First, a complete Literature Review concerning robotic cells and robot calibration is presented. This part allows to improve the knowledge about the works performed during the last years on these subjects. In addition to this, along this Literature Review it is possible to get some ideas of the procedures and the methodologies used in order to design and calibrate a robotic cell.

Secondly, an entire chapter is dedicated to explain the technical details of the design of the robotic cell. This part was time-consuming as it was an optimization job. Many different decisions were made and different scenarios were tested in order to find the most suitable one taking in account the different constraints associated to this robotic cell. This robotic cell must be compact, flexible and collaborative. Flexible means that it must be a system that can be installed and removed with ease, as it will not be a static robotic cell. To be collaborative is the fact that the robots contained inside this robotic cell are prepared to have humans working alongside of the robotic cell without the risk of damage these humans.

Thirdly, a calibration procedure is developed to be applied in this robotic cell. The need of calibration stands because this robotic cell has some strict constraints as compactness and the fact that it is all designed by offline programming. It is crucial to ensure that the robotic cell is installed the closer possible to the scenario decided on the previous chapter. It is clear that it is almost impossible to reproduce in real, the exact same scenario of the numerical simulation software and for this reason the differences must be detected. For this, an automated calibration procedure is implemented in order to adapt the robot paths according to the real positions and orientations of all the components of the robotic cell. This calibration will be performed using a single point laser measurement sensor. Two methodologies were defined for this task:

- One applied to the work-objects based on the cross-product definition;
- Another to define the position and orientation of a cylindrical form solving a linear system of equations.

Finally, a chapter of conclusions is presented with all the relevant aspects of this work and where are also contained the outlooks of this work.

2. LITERATURE REVIEW

This literature review is divided in four sub-chapters. The first three are related to robot calibration as the last one presents the works made to calibrate robotic cells.

It starts with an exposition of the different modelling approaches to process the robot calibration. The following sub-chapter presents some of the measurement systems used to calibrate robots. After it, some works of calibration are analyzed in more detail. The interest on this subject is growing because in the industry domain robots play a main role and the requirement of precise tasks is growing, which means that calibration must be performed with the best possible accuracy. In addition to this, many robotic systems are designed using offline programming software and for this reason calibration is even more necessary.

Finally, the sub-chapter 2.4 explains the procedures made by some researchers in order to perform the calibration of a robotic cell.

2.1. Different modelling approaches

At the beginning, it is necessary to define a mathematical model to represent the error parameters that need to be modelled, establishing the relation between the end-effector poses and the robot joint angles. The most common method is the Denavit-Hartenberg convention [6] [7] [8]. This method consists in defining each link of a robot in relation to the previous one and for each one, four parameters are required. Later, some researchers made some modifications to this model, as Hayati did [9], implementing one more parameter β to consider the deviations when there are two consecutive parallel joints [3].

Other kinematic models are considered in order to improve the error parameter identification such as, the CPC (Complete and Parametrically Continuous) method [8], the POE (Product of Exponentials) method [8] [10] [11] or the Screw-based methods [6] [12]. In [8] it is said that CPC and POE methods are nowadays more complete and makes it easier to perform robot calibration. CPC is a model that includes some modifications from the DH method, meaning that is based on a local link coordinate system while POE method is based on the global coordinate system.

2.2. Measurement systems

One of the most important tasks when doing the robot calibration is the measurement. The accuracy of measurement is very important to succeed in all the calibration process. Comprehensively, the better accuracy wanted, the more expensive the measurement devices are. External devices are used in the majority of the cases of calibration but the tendency is to create autonomous systems of calibration. The measures obtained are used to compare with the theoretical model in order to define the error between the nominal poses and the real measured ones. The measurement system defines if a system follows an open-loop method or a closed-loop method. Open-loop method imposes the use of an external device to measure the pose of the robot end-effector for different robot configurations, while closed-loop means that one Degree Of Freedom (DOF) of the end-effector is constrained by a plane, for example. Some of the external measurement devices are laser trackers [13] [14] like the one seen in Figure 2.1, CMMs (Coordinate Measuring Machine) [15], stereo-vision systems [16] (see Figure 2.2) or measurement arms [17] [18].

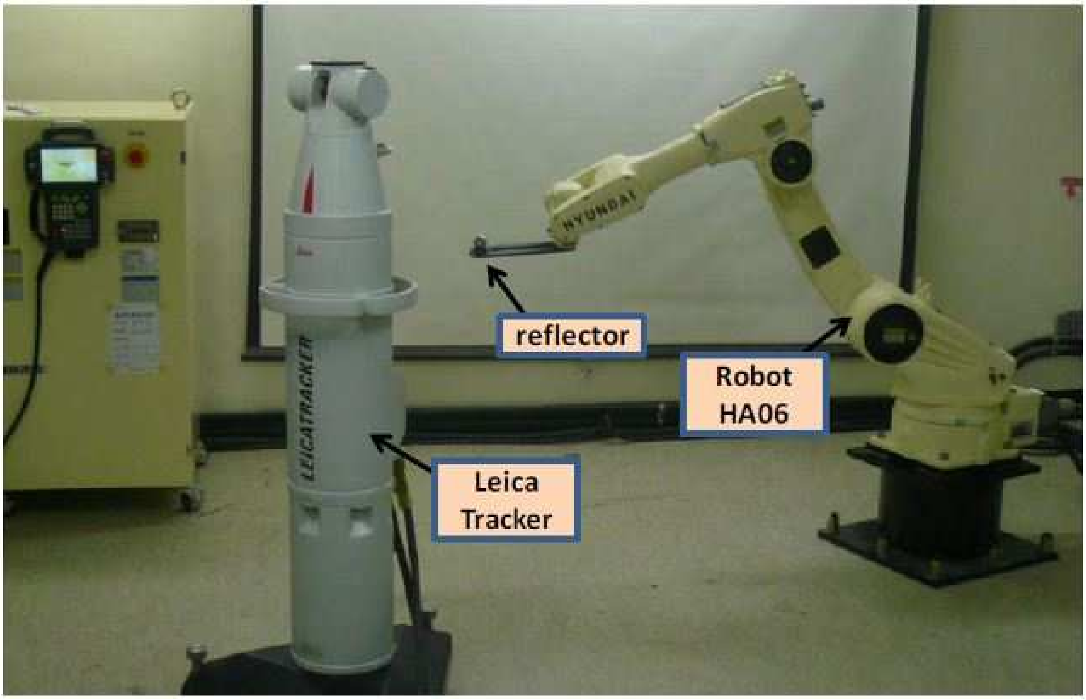


Figure 2.1 Calibration using a Laser Tracker [13]

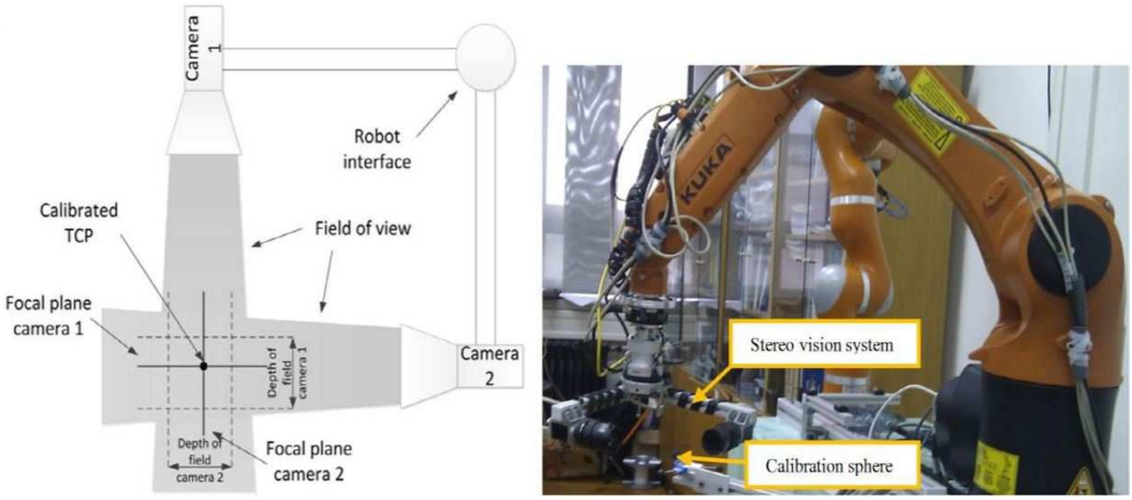


Figure 2.2 Stereo Vision System [16]

On the other hand, other researchers have worked on the improvement of the closed-loop method [19] [20]. Other examples of closed-loop methods are described in two papers which make use of a calibrated block to perform the robot calibration [21] [22]. The last one is presented in Figure 2.3. The block is located in the work space and its position it is well known and the idea is that the robot can touch different planes in different configurations and collect all the data from the robot controller. With this information, the data will be treated in order to obtain the parameters that are defined before to correct the inaccuracy errors of the robot. In both articles only the geometric errors are considered and a kinematic model based in DH is created. In this procedure, 37 different configurations will be measured for each plane. 4 planes are necessary to be touched by the end-effector tool, 3 of them which are mutually orthogonal and the fourth is used to give the scale factor of the measures.

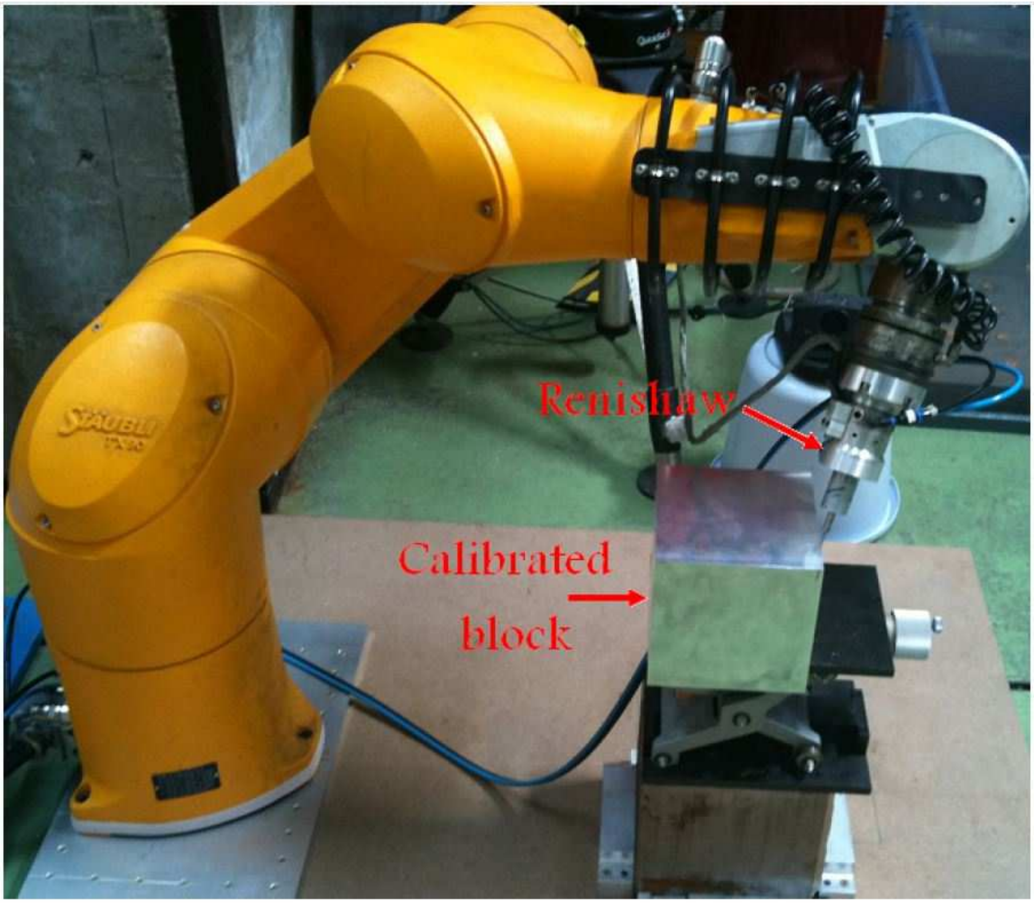


Figure 2.3 Calibration configuration of a Stäubli TX90 [22]

In [23] [24], a telescoping ballbar is linked to a tool fixture that contains three different magnetic cups where the robot end-effector will be attached like it is demonstrated in Figure 2.4. Moving the robot to different configurations will make the ballbar move and those displacements will correspond to measures that applied to a model will give accurate measures of the end-effector poses. This method has the advantage of being more accurate than laser trackers as it is a contact measurement system and it is cheaper than all the different external measuring devices. However, it measures a maximum number of poses limited to 72 which represents an important drawback. This process is successful due to its ease to apply, it is fast and cheap, but the results of this procedure are not yet the expected ones, as the use of the laser tracker produces better results in the whole work space than this system. To sum up, this procedure is perfectly adequate to small industrial robots and when the user wants a “low-cost” system for absolute calibration.

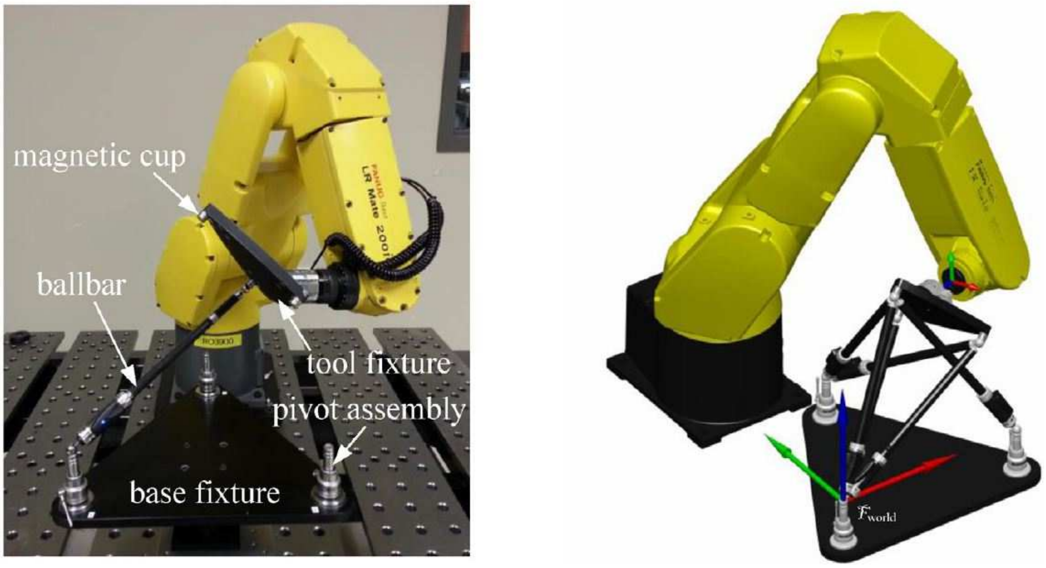


Figure 2.4 Telescoping ballbar system [24]

The authors of articles [25] [26] developed a model where they do not need to measure the complete pose of the end-effector and also a method to obtain an optimal number of configurations in order to avoid of taking unnecessary measures that can even have a prejudicial effect on the results. To prove the efficiency of their model, the results are compared between a full pose measurement system and the partial pose only based in three measurements corresponding to a position. The measures itself are performed with a laser tracker and at the end-effector there is a system containing three positions ready to be measured. With this method they added a step to the calibration process that consists of an algorithm to obtain the optimal number of measures needed to obtain the best results in the calibration process. This new step was previously been worked by other researchers [27]. They called this step the *design of experiments* and it is performed just before the measurement step. In Figure 2.5 it is presented a setup as example to apply this new procedure.

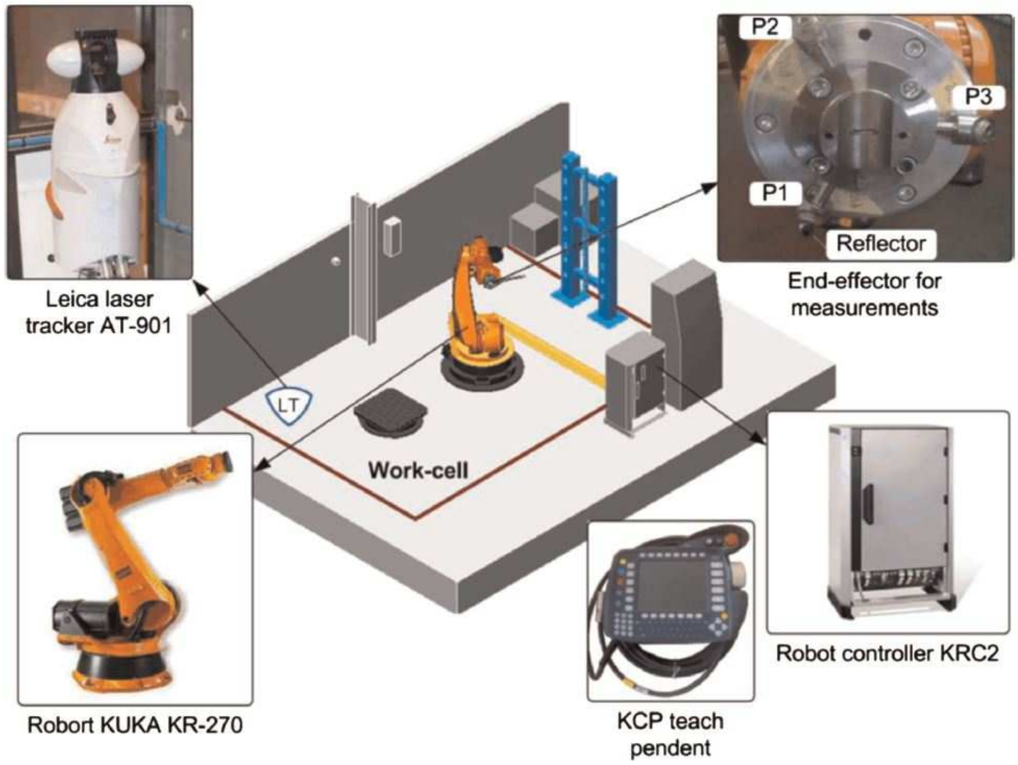


Figure 2.5 Experimental setup for manipulator geometric calibration [26]

2.2.1. Autonomous calibration systems

The new tendency as said before is to apply systems to the robots itself in order to transform the calibration procedure into an autonomous process. Laser measurement systems and cameras are the vision systems that almost all the autonomous systems use. Associated to this vision systems usually there is also a board with black and white squares that is used as a relative measurement system. This is called a *calibration board*. The system presented in Figure 2.6 is a good example of this different calibration method [28].

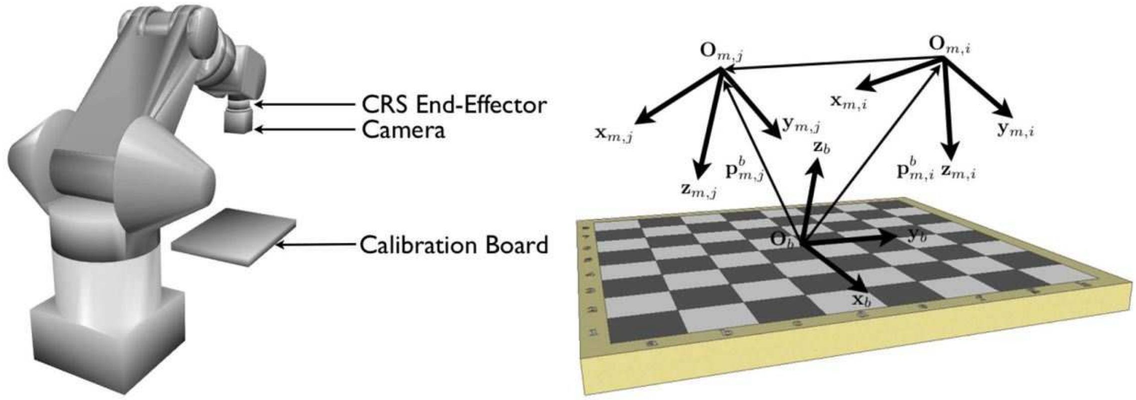


Figure 2.6 Relative measurement using a camera and a calibration board [28]

This system is based on image analysis. The camera is taking pictures pointing the calibration board and when the robot changes its position or configuration it is possible to have another image. Comparing both, with the difference between theoretical and real end-effector poses is possible to process the calibration.

Another system that can be considered approximate to this is the one in [29]. The difference is that in this one also the nongeometric effects are taken into account.

The last two papers cited use a camera to process the calibration and in [30] a CCD camera is also used to adapt the virtual world of an offline programming model to an actual robot. This system is shown in Figure 2.7. This work differs from the others because there is no need of a specific camera and even the camera also does not need any calibration. In this model, in order to identify the camera view line, the robot motion will be autonomously created. In this work, to evaluate the accuracy of the position measurement, also a plate with defined targets is used.

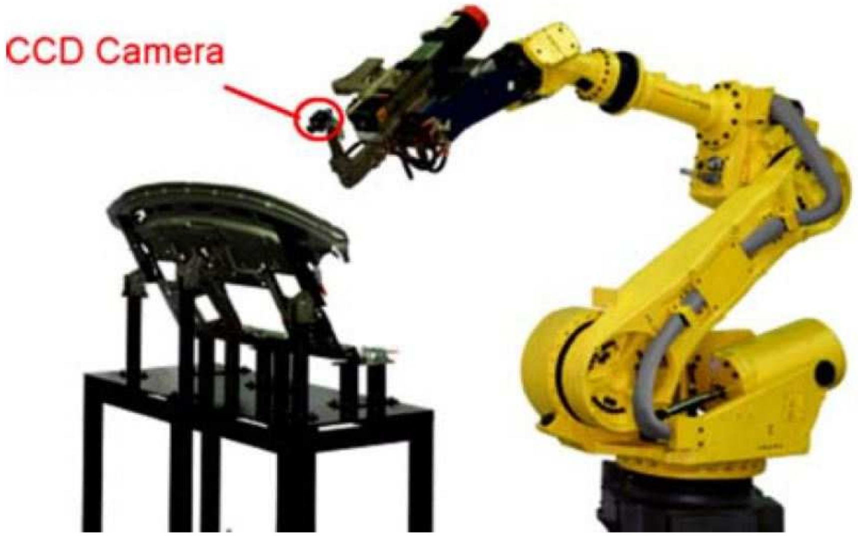


Figure 2.7 Experimental robot system with a CCD Camera [30]

In order to develop the autonomous calibration systems to industrial robots, Nieves et al. [31] [32] [33] proposed a new calibration method where they apply a laser to the robot tool center point (TCP) and place a portable position sensitive device with two fixed position sensitive detectors (PSD’s). The method consists in teaching the robot controller 4 positions that makes the laser aim one of the PSD’s and there it is reflected to the other one. Two positions are pointing one PSD and the other two are aiming the second PSD (see Figure 2.8). Analyzing the recorded joint angles of the robot controller and applying the robot forward kinematics it is possible to develop a new calibration method.

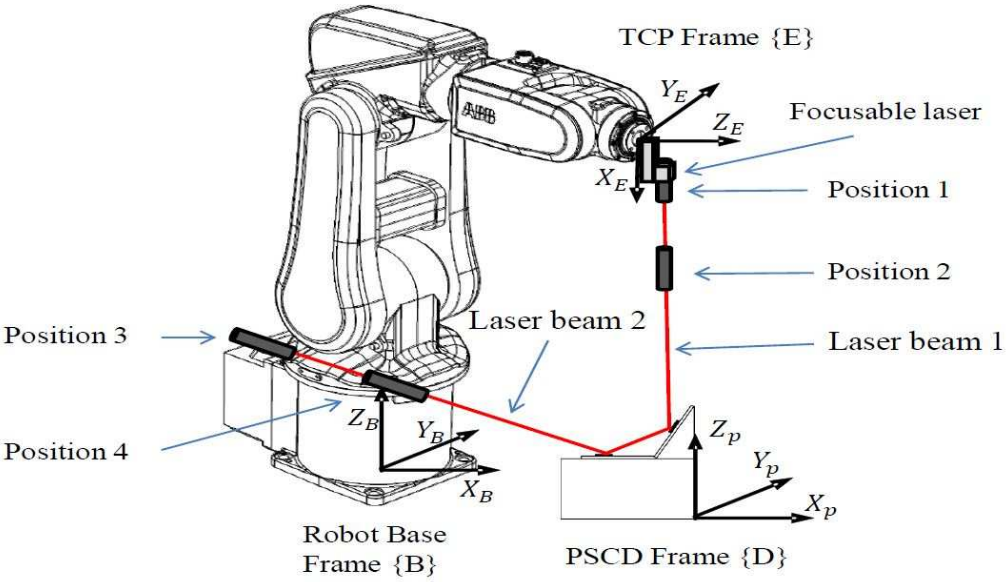


Figure 2.8 Calibration system with Position Sensitive Detectors [33]

2.3. Analysis of some previous works

In 1997 [34] the robot calibration was already being developed. In this work to obtain the geometrical parameters of the robot, the DH convention was used. The measuring system used is a system with three wires and the lengths of each are continuously being measured. This system is called ROBOTRAK. The vector of errors (R) in the robot are expressed as follows:

$$R = M(x, \theta) - Y \tag{1}$$

Where M represents the model position and Y represents the position given by ROBOTRAK. x is the vector of model parameter coefficients and θ is the vector of joint angles.

Each element of the vector R represented in Equation (2) equals the Euclidean distance between the robot end-effector position given by ROBOTRAK and the position calculated using the model.

$$r_i = \sqrt{\Delta x_i^2 + \Delta y_i^2 + \Delta z_i^2} \tag{2}$$

To make the correction of the parameters the method used was the Levenberg-Marquardt. The algorithm is presented below (3). This iterative method will stop when three successive iterations does not change by a value larger than $10^{-4}\ mm$. The minimum value of the function is then reached.

$$X_{k+1} = X_k - [J(X_k)^T \cdot J(X_k) + \lambda_k \cdot I]^{-1} \cdot J(X_k) \cdot R(X_k) \tag{3}$$

X_k: Vector of model parameters;
k: Iteration number;
R_k: Vector of residuals;
$J(X_k)$: Jacobian matrix of the partial derivates of $R(X_k)$ with respect to X_k;
I: Identity matrix (29, 29).

The use of this model was successful when it was published. The average error improved from 17.88 mm to only 1.16 mm.

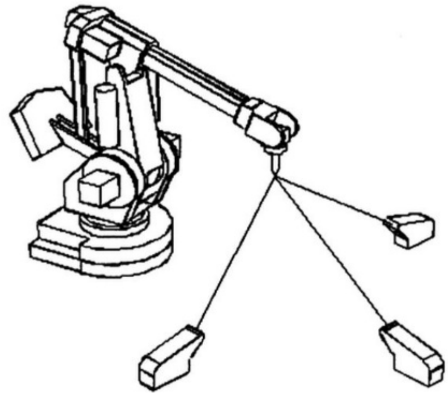


Figure 2.9 ROBOTRAK calibration system [34]

Gong et al. [1] developed a complete model of robot calibration with special attention dedicated to the nongeometric errors. Those errors were divided in two categories: link compliance and thermal effects. Also the geometric parameters were corrected but will not be analyzed here. The formulation of the model used is the following:

$$\Delta X = J\Delta P \tag{4}$$

Where:

$$\Delta X = \delta X_G + \delta X_C + \delta X_T \tag{5}$$

ΔX: represents the global position and orientation errors;
δX_G: end-effector error due to geometric errors;

δX_C: end-effector error due to compliance errors;

δX_T: end-effector error due to thermal errors.

The parameters used to identify the errors to modify are based in the DH modified convention which contain the β parameter by Hayati [9].

$$\Delta P = (\Delta\theta\ \Delta d\ \Delta a\ \Delta\alpha\ \Delta\beta) = \delta P_G + \delta P_C + \delta P_T \tag{6}$$

In detail:

$$\delta P_G = (\delta\theta_G\ \delta d_G\ \delta a_G\ \delta\alpha_G\ \delta\beta_G)^T \tag{7}$$

$$\delta P_C = (\delta\theta_C\ \delta d_C\ \delta a_C\ \delta\alpha_C\ \delta\beta_C)^T \tag{8}$$

$$\delta P_T = (\delta\theta_T\ \delta d_T\ \delta a_T\ \delta\alpha_T\ \delta\beta_T)^T \tag{9}$$

δP_G: parameter errors due to geometric influence;

δP_C: parameter errors due to compliance;

δP_T: parameter errors induced by thermal errors.

The Jacobian matrix is composed by the partial derivatives of the end-effector position (M_i) and orientation (R_i) with respect to the kinematic parameter errors:

$$J = \begin{pmatrix} M_\theta & M_d & M_a & M_\alpha & M_\beta \\ R_\theta & 0 & 0 & R_\alpha & R_\beta \end{pmatrix} \tag{10}$$

In this calibration process, compliance errors generated by joint deflection are much more significant when compared to the ones generated by link deflection. This allows to establish a relationship between the differential motion of the end-effector (δX_c) and the small change of the joint parameters ($\delta\theta_C$).

$$\delta X_c = J_\theta \delta\theta_C \tag{11}$$

Simply by applying the torque definition (τ) and obtaining the small differential of the joint parameters it is possible to find the differential of the end-effector pose. Assuming that the joints act as linear torsional springs, a linear constant can be used to represent the compliance of each joint (C_θ^*).

$$\delta\theta_C = C_\theta^* \tau \tag{12}$$

An example was applied to a 6 Degrees of Freedom (DOF) robot considering only joints 2 and 3, which are the joints where most of the compliant errors due to the gravity occur. Also joint 5 is in the same relative position which means that errors are also significant in that joint, but in this article only the other two were considered. This makes possible to obtain three dimensionless parameters that depend on the weight of the links, the distance between the joint axis and the gravity center, and the compliance parameters (C_θ). As this information is not easy to find, using the inverse kinematics will allow to estimate this three parameters.

Using an external measurement system (Laser Tracker) it is possible to obtain the matrix with the actual end-effector position and orientation (X). This measures will then be compared to the desired end-effector position and orientation (X') given by the robot controller. Using equation (4) as the basis for the kinematic parameter identification, the least square solution for ΔP is given by the following equation:

$$\Delta P = (J^T J)^{-1} J^T \Delta X \tag{13}$$

Where:

$$\Delta X = X - X' \tag{14}$$

First, the geometric (δP_G) and the compliance (δP_C) errors will be calibrated considering that in the environment there is no temperature variation. Then, to understand the thermal effects, the robot will warm up by exercising itself at high speed and then cool down. As it is seen in Equation (4), the relationship between the end-effector positioning accuracy and the individual parameter error is linear. Considering that the parameter errors are small, the thermal errors can then be expressed through Equation (15).

$$\delta P_T = \Delta P - \delta P_G - \delta P_C \tag{15}$$

In this work, thermistor sensors are used to acquire the temperature. Eleven were placed in different locations of the robot structure and another one was placed in the space to monitor the environment temperature.

In calibration one of the ways is to calibrate the base frame and obtain the pose described in the form of a 4x4 homogeneous matrix, containing the 3x3 orthogonal rotation matrix. One work [35] was performed to improve this procedure, particularly the orthogonality of the robot base frame's rotation matrix using the unit quaternion form. In this work the calibration is divided in two steps: preliminary and fine calibration. In the preliminary calibration process 5 TCPs were measured instead of only 4 which resulted in an improvement in limiting the influence of the measuring error. In the fine calibration step, the quaternion form was used in order to improve the measurement accuracy of the actual pose of the robot base frame.

Another type of calibration [36] uses a camera and a laser pointer attached to the robot end-effector like illustrated in Figure 2.10. The laser pointer points to a surface and the camera which is located over the robot work space detects the shape of the laser spot in the surface and with it obtain the orientation of the robot end-effector. The laser spot position is sensitive to the orientation of the end-effector due to the extensibility of laser beam. It is necessary to know the transformation matrix between the laser pointer and the robot base frame (${}^{b}H_{L}$), and between the robot base frame and the camera (${}^{c}H_{b}$).

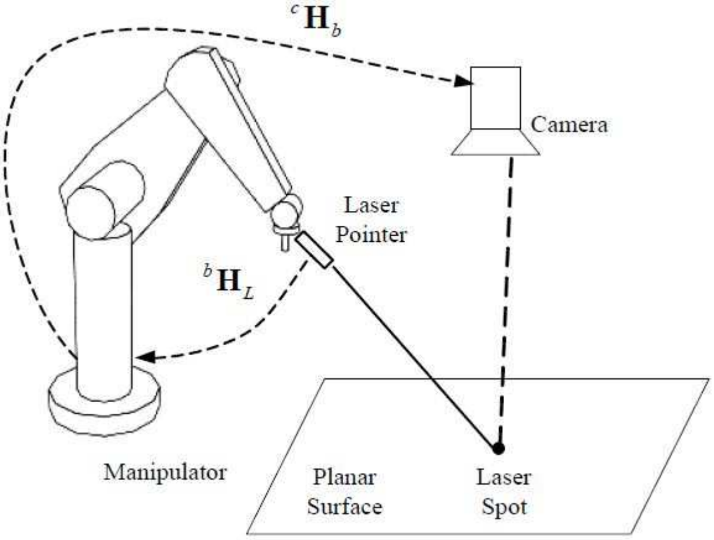


Figure 2.10 Laser pointer and camera system with its coordinate relationship [36]

2.4. Robotic cells

A robotic cell is a system where there are robots, tools, work pieces, conveyors or structures dedicated to a specific task. To perform the calibration of an entire robotic cell, several parameters must be taken into account. When a robotic cell is installed a procedure must take place to identify the real position of all the components in the cell. A robotic cell can previously be modelled using offline programing. It allows to evaluate the feasibility of the system trajectories and to test different configurations to place all the components. This procedure allows to project the best solution and is used to avoid, if possible, errors when testing the real situation of the robotic cell. The real simulation must always be performed, this is why it is so important to minimize the errors that can always appear when testing and building a new system. A system like this must be prepared to recognize the positions of all the components composing the cell. Several frames and relationships between them must be defined (see Figure 2.11).

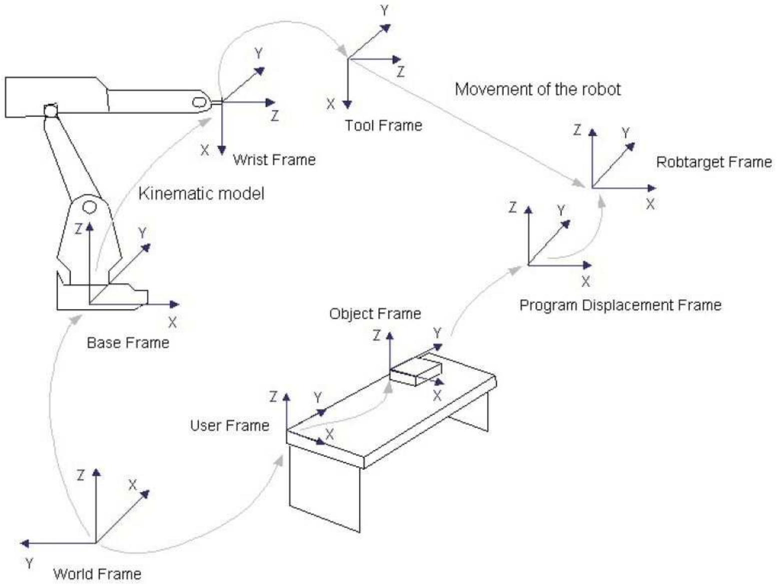


Figure 2.11 Different frames in a robotic cell [37]

It is said that the calibration of a robot cell can be divided into six groups [38]:

- Absolute Accuracy Calibration: calibration of each of the mechanism’s kinematics;
- Home Calibration: calibration of each component base frame;
- Cell Alignment: obtain the relative positions between the components in the robotic cell;
- TCP Calibration: get the transformation between the tool center point and each of the manipulators;

- Work-object Calibration: get the positions and relations between the work pieces and the other components;
- Process Fine Tuning: overall verification of all the calibration steps to minimize errors that could have occurred before.

Li et al. [37] described in an article some ways of calibrating a robotic cell. It can be performed by a sensor. This sensor can be stationary to measure the object that is held by the robot or it can be attached to the robot end-effector to measure a fixed object. Then it is also suggested to use a trigger and a sphere. Six different approaches with this calibration objects can be performed in order to calibrate the robotic cell.

In a robotic cell where there are more than one robot it is important to establish a relationship between both robots [39] [40]. As the robots are immobile, the base frames are static and it is possible to define the relation between both the robot base frames. The following step is to define the relations: hand-eye and tool-flange. Usually this step is performed independently using $AX = XB$ and $AX = YB$, where A, B, X, Y are homogeneous transformation matrices of end-effector movements, camera movements, hand-eye transformation and the robot to world rigid transformation, respectively. In [39] a new approach to a robotic cell is performed. In order to reduce the initial data for calculation and to have better results, they used a $AXB = YCZ$ formulation to calibrate the robotic cell (C stands for the second robot end-effector movements and Z represents the tool-flange transformation). This simultaneous model is then compared with the 3-step and 2-step methods. The 3-step method is performed obtaining X, Z and Y separately. X and Z follow the $AX = XB$ formulation as to obtain Y it is used the least squares method using the previously retrieved data. The 2-step method obtains 2 parameters with the $AX = YB$ formulation as Z is obtained using the $AZ = YB$ formulation. These 3 different methods are compared and the better results obtained are the ones using the simultaneous method instead of the non-simultaneous methods (3-Step and 2-Step methods). Results were presented in terms of average accuracy and stability against noise. However, the efficiency of this simultaneous method is strongly dependent on the initial guess defined. Considering that the relationship between the hand-eye and tool-flange is difficult to obtain, an appropriate methodology is necessary to reach an approximate initial guess to the iterative algorithm.

To sum up, the hand-eye, tool-flange and robot-robot relationships have to be determined frequently in order to enable the robots to cooperate inside the constantly changing environment.

Last year [41], an interesting method of calibration is used in a robotic cell (see Figure 2.12). To perform this calibration it is only needed a structure with a ball and a cube. This structure will be placed in a position where the robot can touch them with a displacement sensor that will be attached to the robot end-effector.

To calibrate the tool offset it is necessary to touch the ball in different locations and then it is possible to obtain with the least square method the position of the tool in relation to the base reference frame. The method used is the Levenberg-Marquardt and as it is a non-linear algorithm, it is important to make a proper estimation of the initial value. In this work, this initial value used is the nominal value taken from the CAD model.

Figure 2.12 Calibration structure with a ball and a cube [41]

The second part of this work is to calibrate the work object, which means that it is necessary to precisely know the location of the work object in the work space. For this approach, the cube is used to obtain three adjacent planes. First, it is necessary to fit a plane with the points measured in each cubic surface. Secondly, after performing it in three adjacent planes it is necessary to obtain the intersection point. It can be obtained either by solving the linear equation with the three plane equations or by least square using the three normal directions and points in each of the three planes. Finally, the orientation must be calculated. The conventional way is to use two of the three normal vectors obtained before and with it obtain the third one. But in this work a cyclic cross method is proposed (Figure

2.13). They use the conventional way but this is performed six times. It is performed to all the combinations possible between the based vectors and the deduced one. This means that six different orientation frames are obtained and an average is then calculated from the six frames. This method improves the accuracy of the orientation calculation.

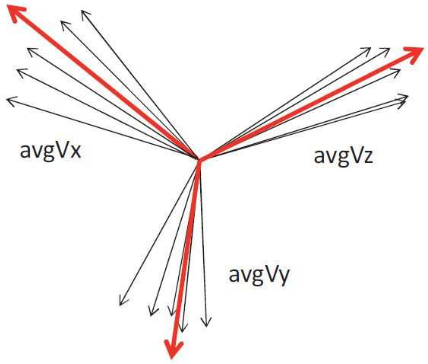


Figure 2.13 Illustration of cyclic cross method [41]

Another procedure to calibrate a robotic cell is the use of DynaCal Robot Cell Calibration System from Dynalog, Inc. which allows to calibrate the robot, the end-effector and the work piece in a robotic cell simultaneously using a set of static position measurements [42]. This method is used to easily correct and compensate the errors detected by the measurement system presented in Figure 2.14.

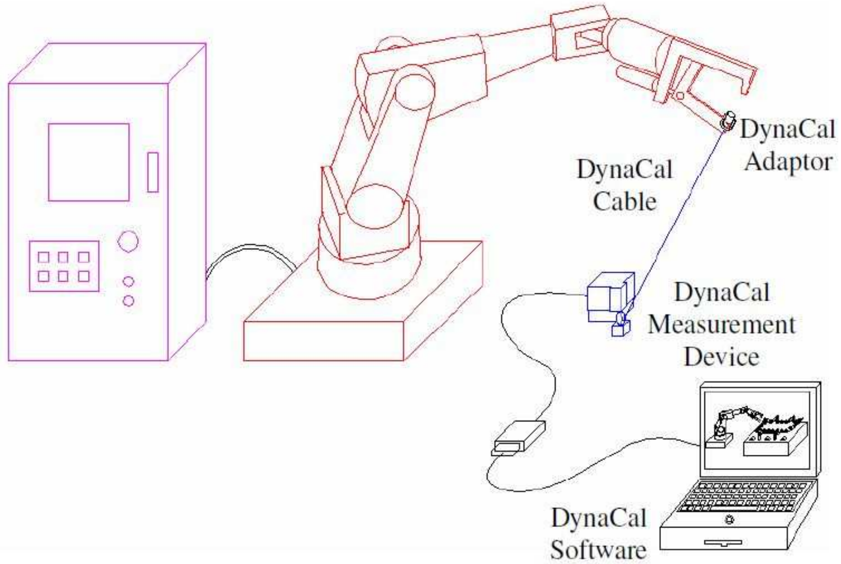


Figure 2.14 DynaCal Robot Cell Calibration System [42]

Sometimes the problem when calibrating a robotic cell is that the process only covers and calibrates for certain positions of the end-effector. In [43] a system tries to be adapted to make possible to calibrate more positions and configurations of the robot. The calibration structure is presented below:

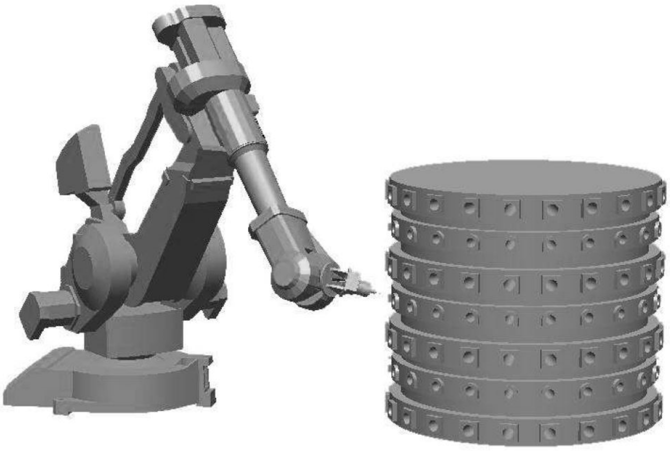


Figure 2.15 Calibration structure to cover the majority of the workspace [43]

This literature review gives a good idea of the works performed during the last years concerning robot calibration. In addition to this, it also presents the latest robotic cell calibration papers, from which some methods of designing and calibrating a robotic cell were taken in account to the work performed during this Master thesis.

3. ROBOTIC CELL DESIGN

The design of a multi-robot work cell has many aspects to concern about. The first step is to know the tasks that are supposed to be performed. This will allow to define all the components necessary inside the robotic cell. It is important to have an idea of the dimensions and weights of the components in order to choose the most suitable robots to perform the desired tasks. When choosing a robot, we must mainly see the rated payload, its maximum reach and its flexibility (light-weighted, easy to displace). For this particular robotic cell, the robots used must be collaborative and flexible. These are the main decisions at the beginning when designing a complete new robotic system.

When the components and the robots are defined, it is time to start to decide the positions and orientations of the different components in the robotic cell. This part is time-consuming as all the different scenarios must be considered in order to find the most suitable one at the end. This part must be performed with a numerical simulation software to be possible to see the advantages and drawbacks between the different scenarios tested.

V-REP (Virtual Robot Experimentation Platform) was the software used to perform all the robotic cell simulations. This is a free software that allows to import robots to a scene and perform simulations with them. It is also possible to import CAD-files in ".stl" format. This software has many useful functions such as: collision detection, distance calculation, motion and path planning.

3.1. Architecture conception

This robotic cell is composed by:

- 2 robots *KUKA LBR IIWA 14 R820* (see Figure 3.1);

[redacted]

Figure 3.1 ***KUKA LBR IIWA 14 R820***

These robots' rated payload is 14 kg, its maximum reach is 820 mm and it weights approximately 30 kg. The choice of the robot in Figure 1 is due to its main characteristic which is to be collaborative. Being collaborative means that it is prepared to work alongside with people without the danger of strongly hit someone. This is only possible because this robot allows to control the torque applied on each of its 7 axis. This allows to detect any contact with the robot making him stop immediately. It is also useful as a detection tool. For example, a work piece coming on a conveyor touches the robot to initialize the pre-established process. In addition to this, it is a robot that is easy to handle and remove from one site to another as it only weights 30 kg.

The aim of this robotic cell is [redacted] This procedure will be divided in 7 different tasks performed by the robots:

- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]

The need of two robots for this robotic cell is due to the weight of [redacted] [redacted]. Its weights are heavier than 14 kg and these tools must be carried by the two robots in collaboration. Here are the weights of the tools inside this robotic cell:

- [redacted]
- [redacted]
- [redacted]
- [redacted]
- [redacted]

3.2. Development of the Robotic Cell

To create a robotic cell it is extremely important to have in mind the demands of the enterprise and also the restrictions associated to each object in the robotic cell in order to reduce the number of different possible scenarios. [redacted]

[redacted]

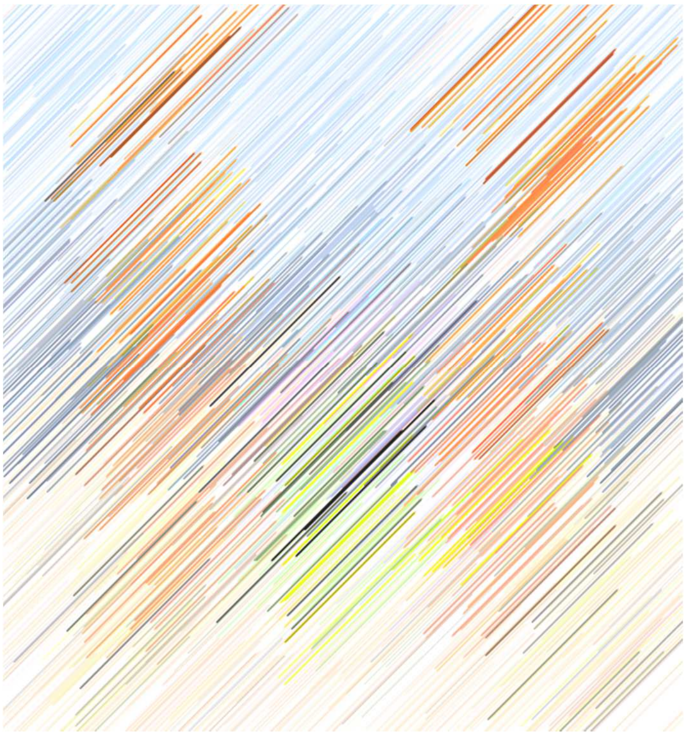


Figure 3.2 [redacted]

3.2.1. Height of the robot bases

[redacted]

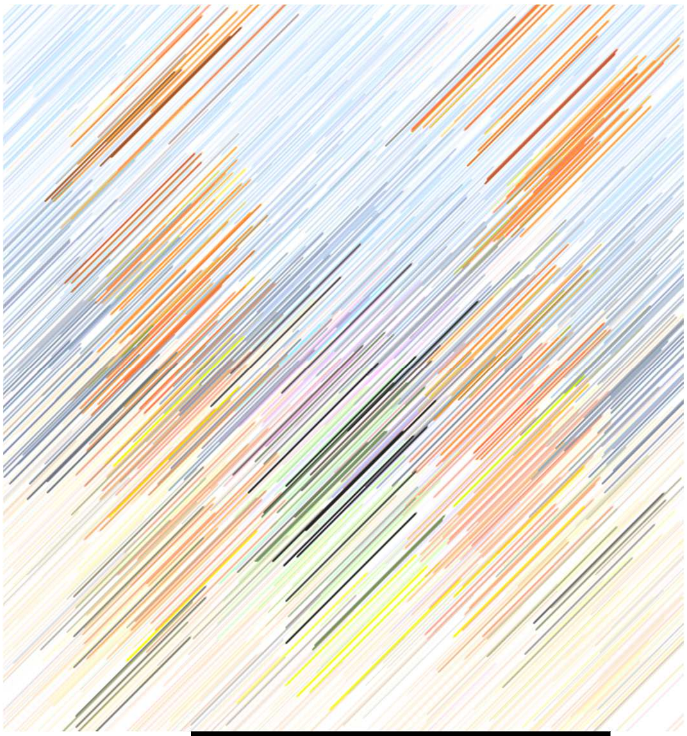


Figure 3.3 [redacted]

3.2.2. Distance between the different components

It is also necessary to define the distance of the robots to the reference point. It should be enough to allow the robots make its tasks between them without the risk of collision and can't be excessive because it is supposed to be a compact robotic cell and also regarding the robots' maximum reach. [redacted]

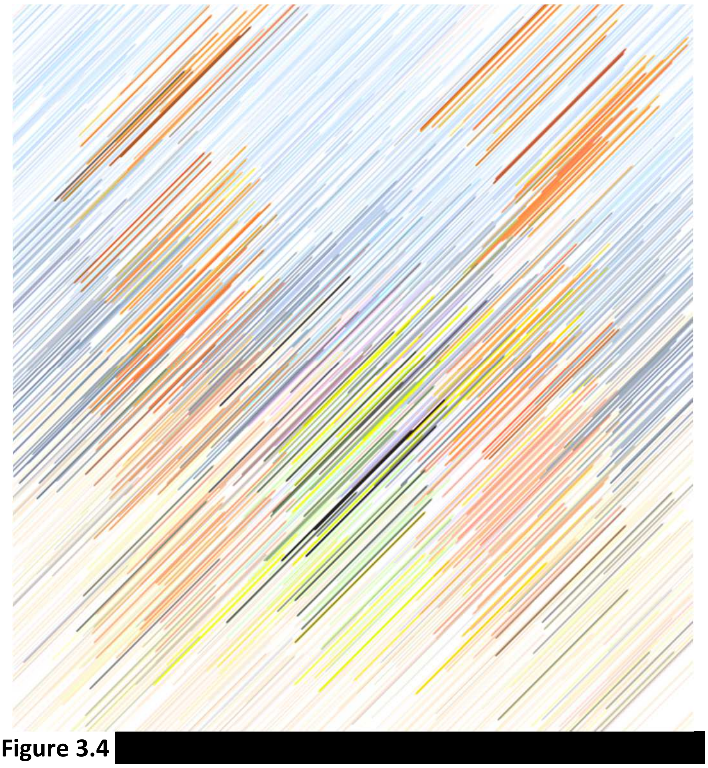


Figure 3.4 [redacted]

[redacted]

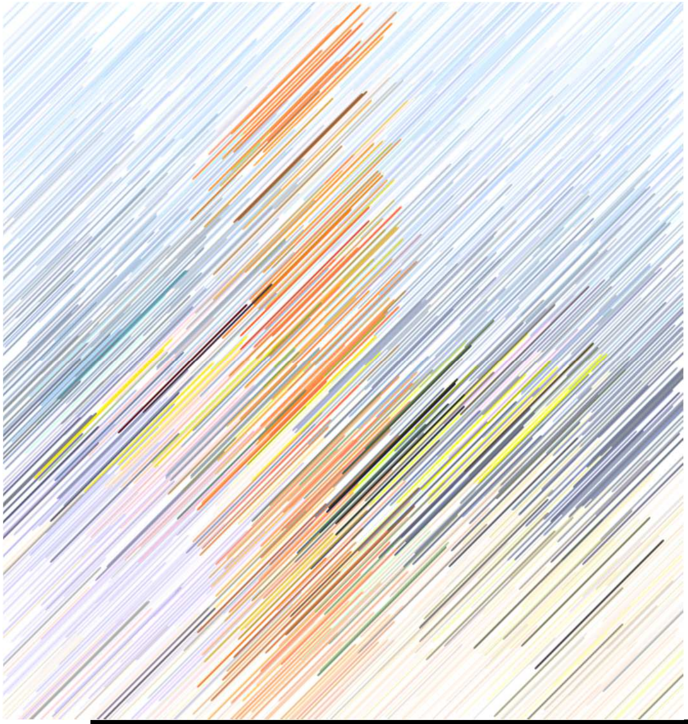


Figure 3.5 [redacted]

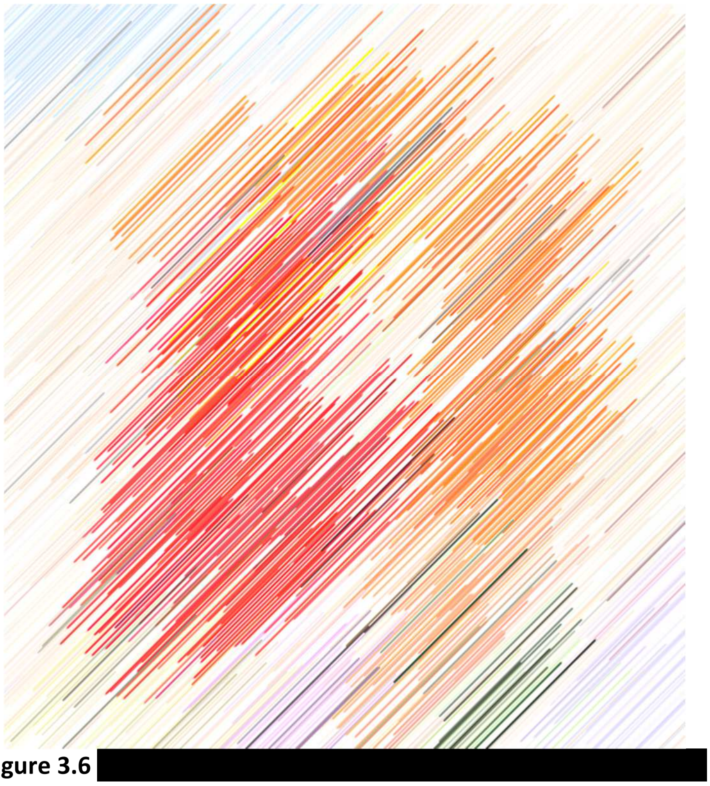


Figure 3.6 [redacted]

3.2.3. Disposition of the tools inside the robotic cell

The tools of this robotic cell were placed according to the disposition presented in Figure 3.7:

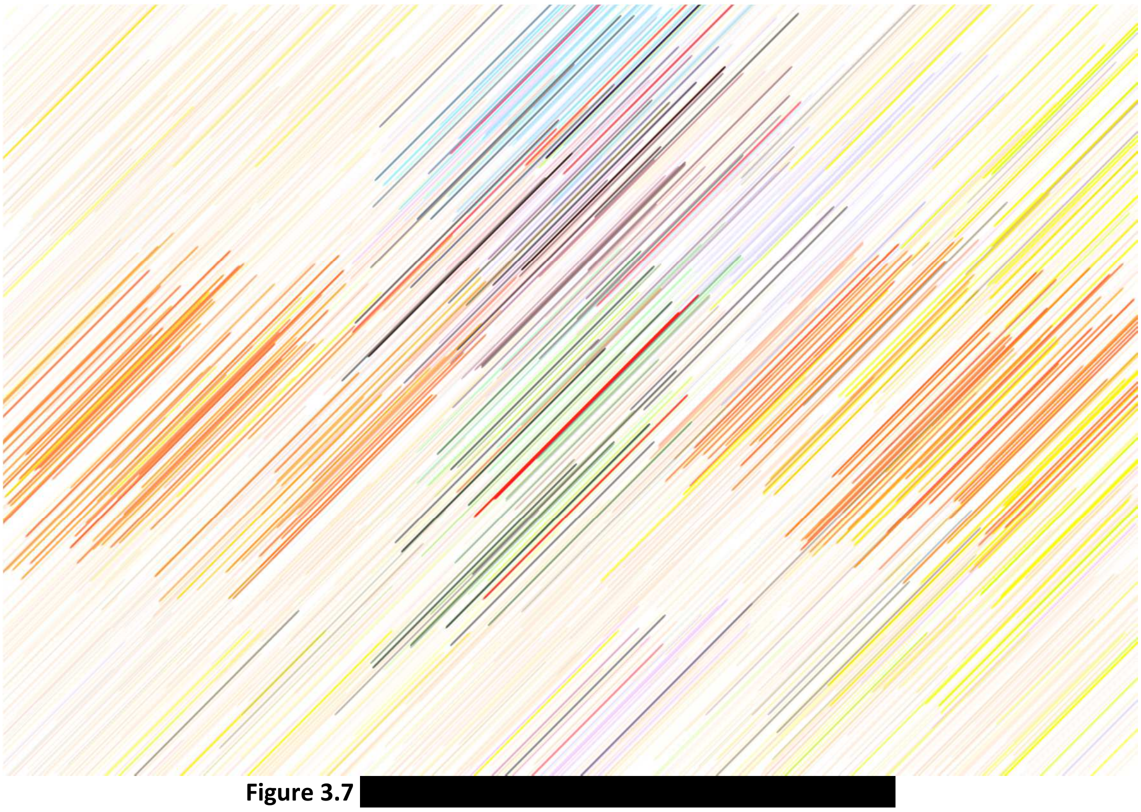


Figure 3.7 [redacted]

To decide the position and orientation of all the tools, it was necessary to have in mind the task performed by the tool and also the way of carrying that tool (either by one single robot or by both of them in co-manipulation). [redacted]

[redacted]

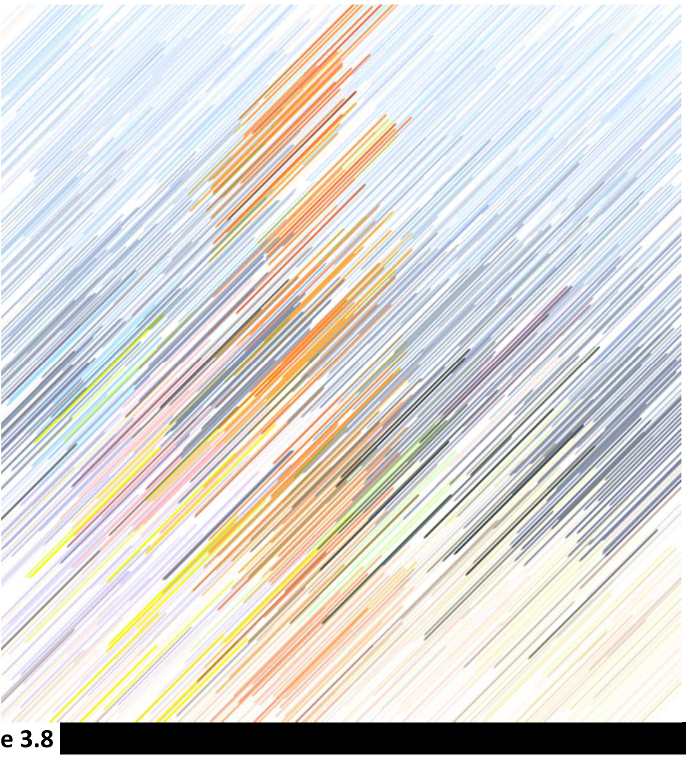


Figure 3.8 [redacted]

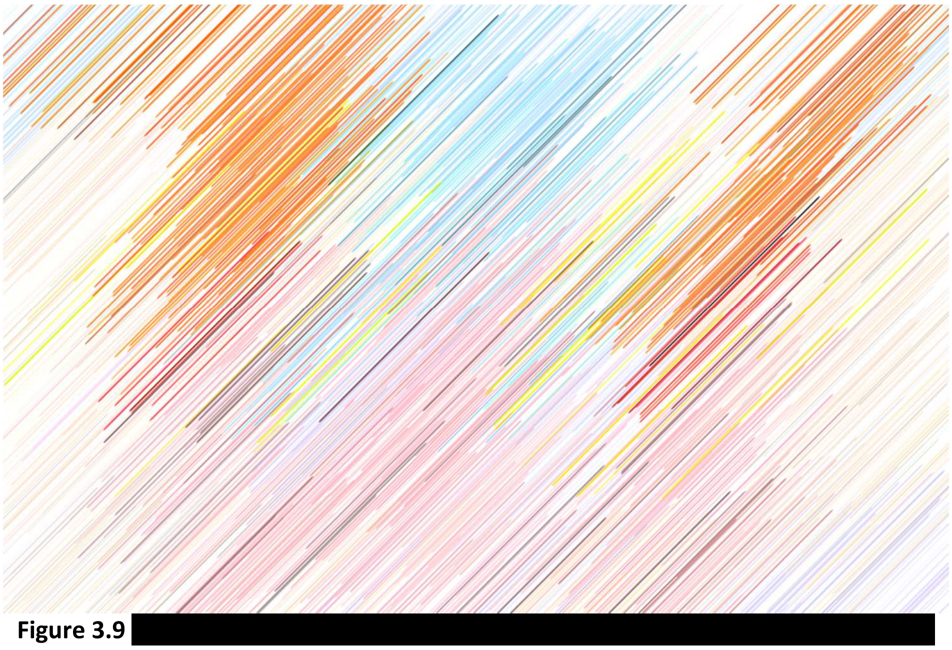


Figure 3.9 [redacted]

[redacted]

[redacted]

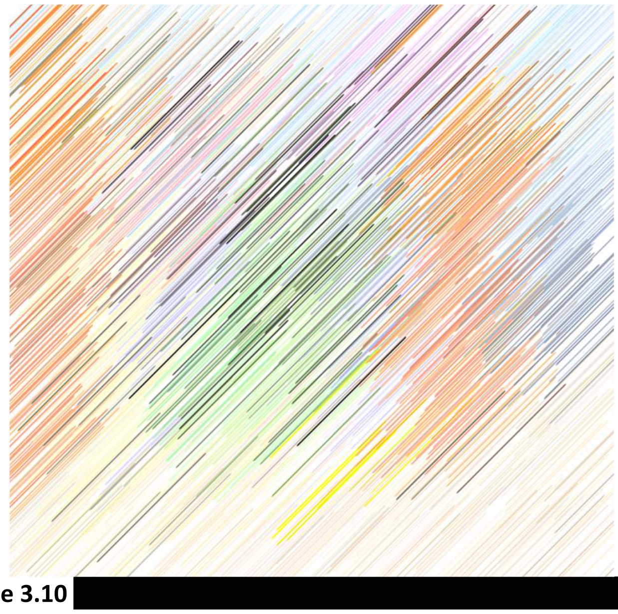


Figure 3.10 [redacted]

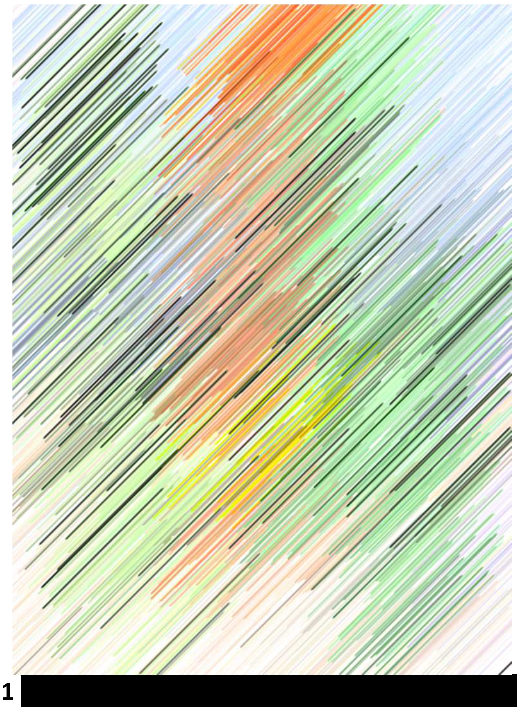


Figure 3.11 [redacted]

3.2.4. Degrees of freedom of the robotic cell

After defining the architecture that was according to all the demands and constraints of this robotic cell, some tests were performed in order to have the degrees of freedom of displacement of certain components of the cell. [redacted]

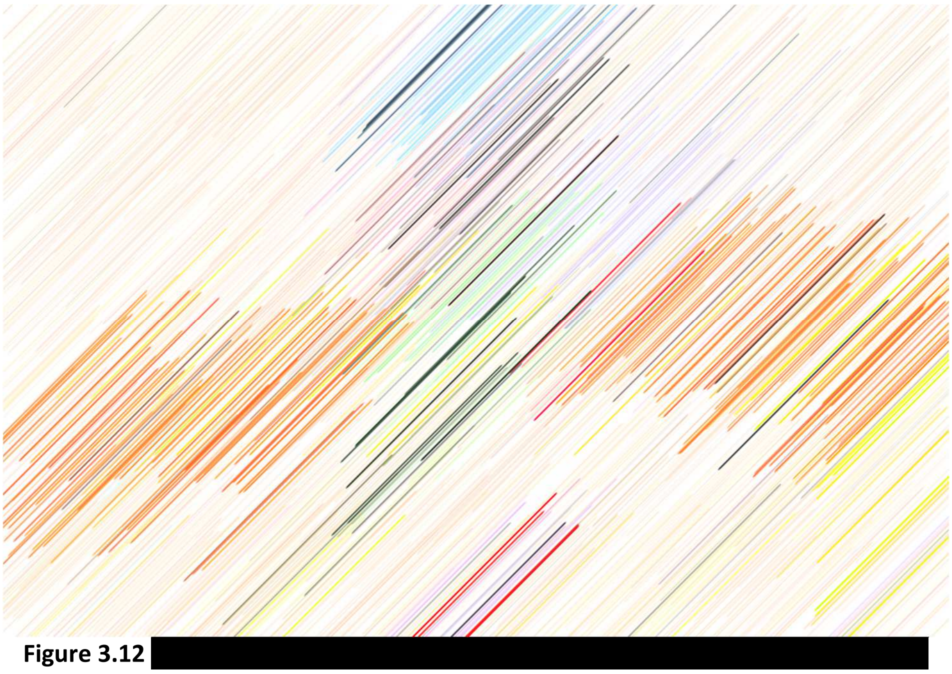


Figure 3.12 [redacted]

[redacted]

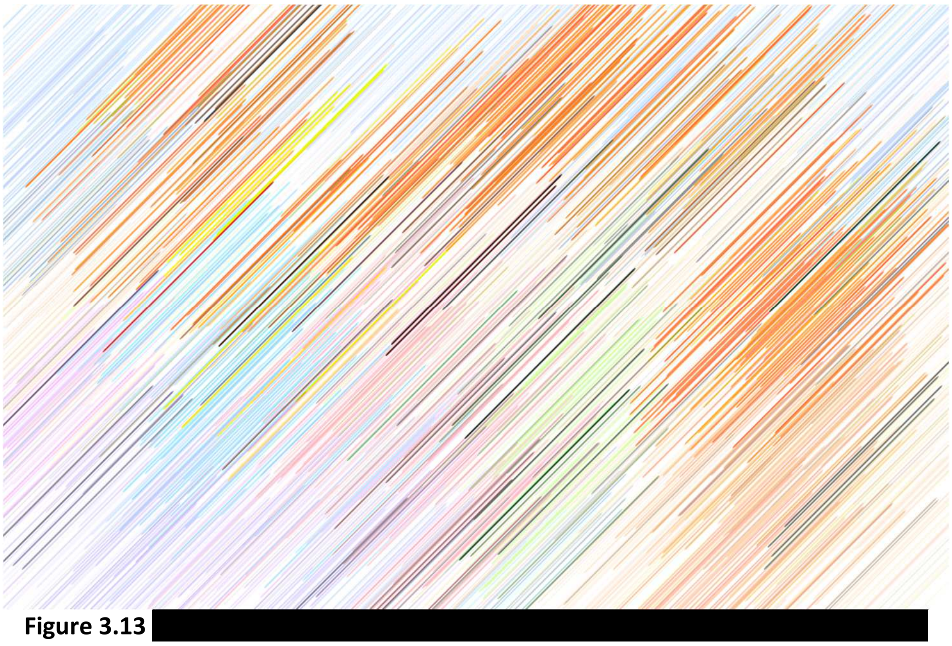


Figure 3.13 [redacted]

[redacted]

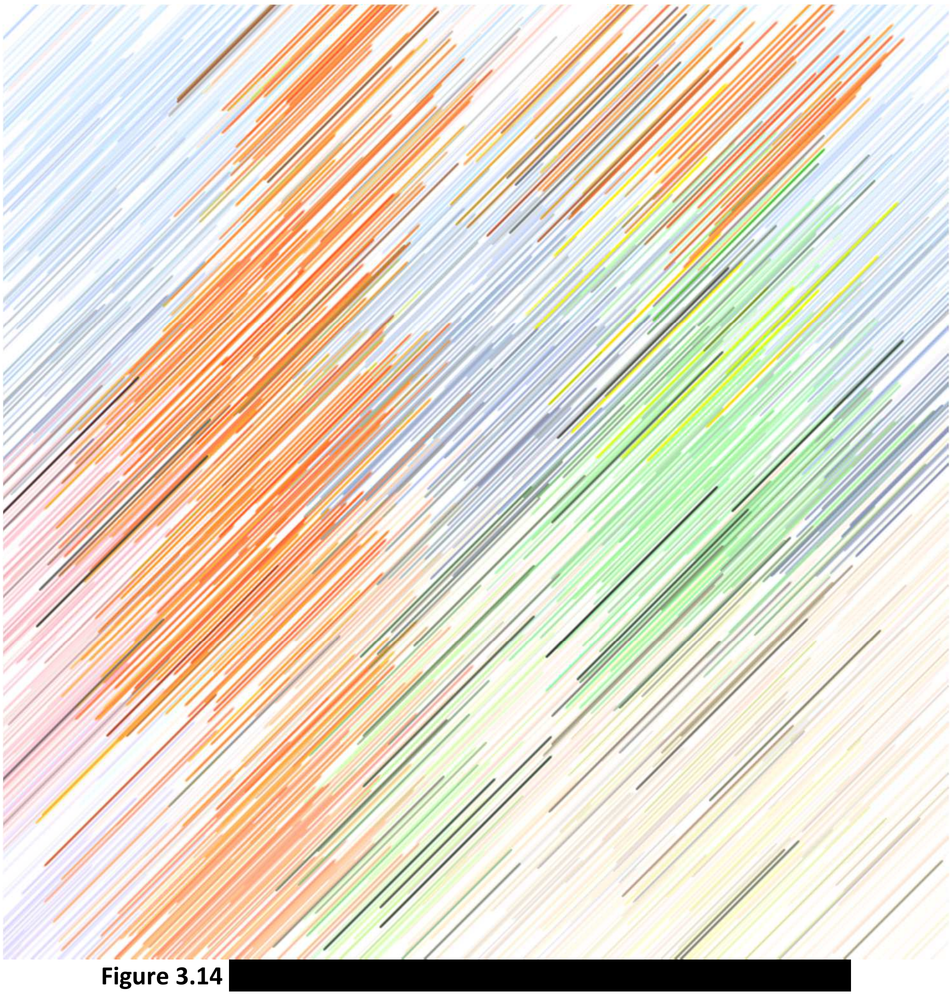


Figure 3.14 [redacted]

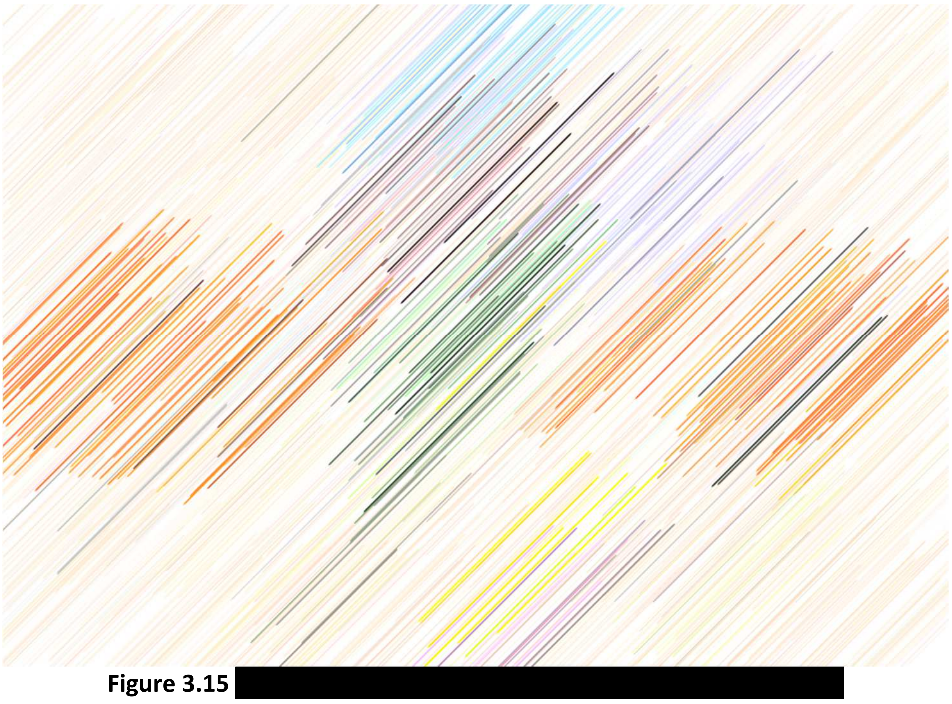


Figure 3.15 [redacted]

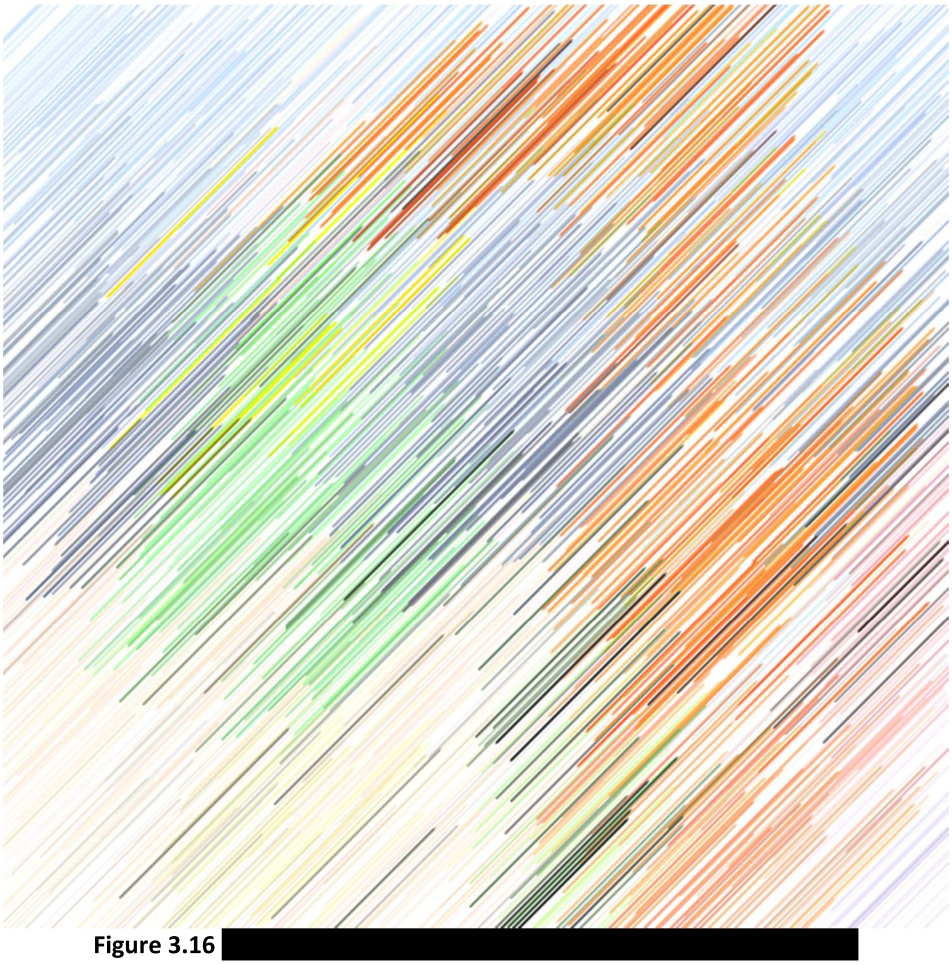


Figure 3.16 [redacted]

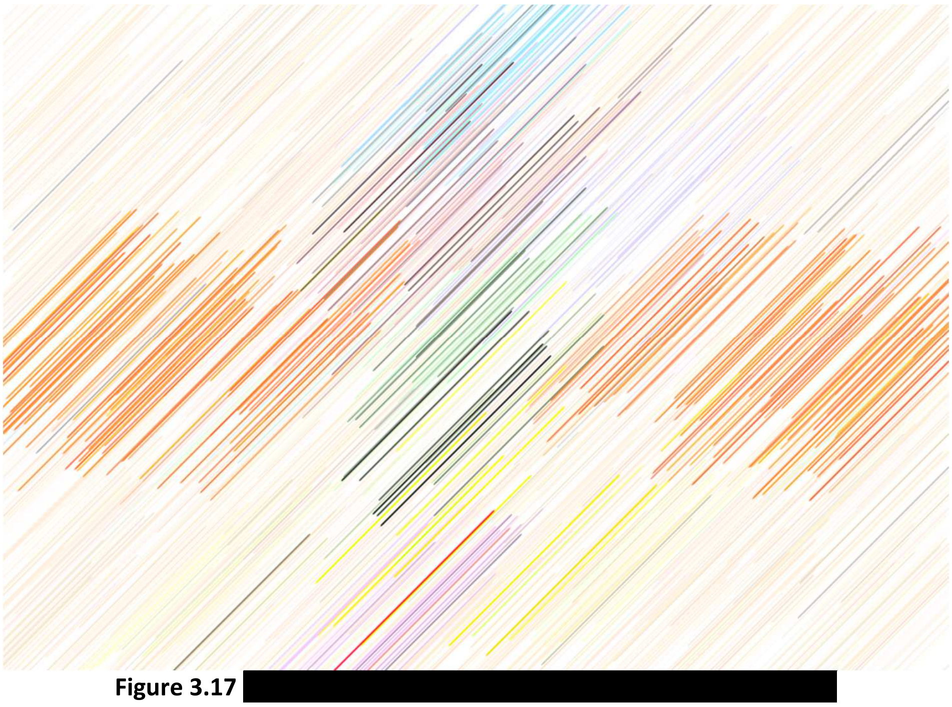


Figure 3.17 [redacted]

[redacted]

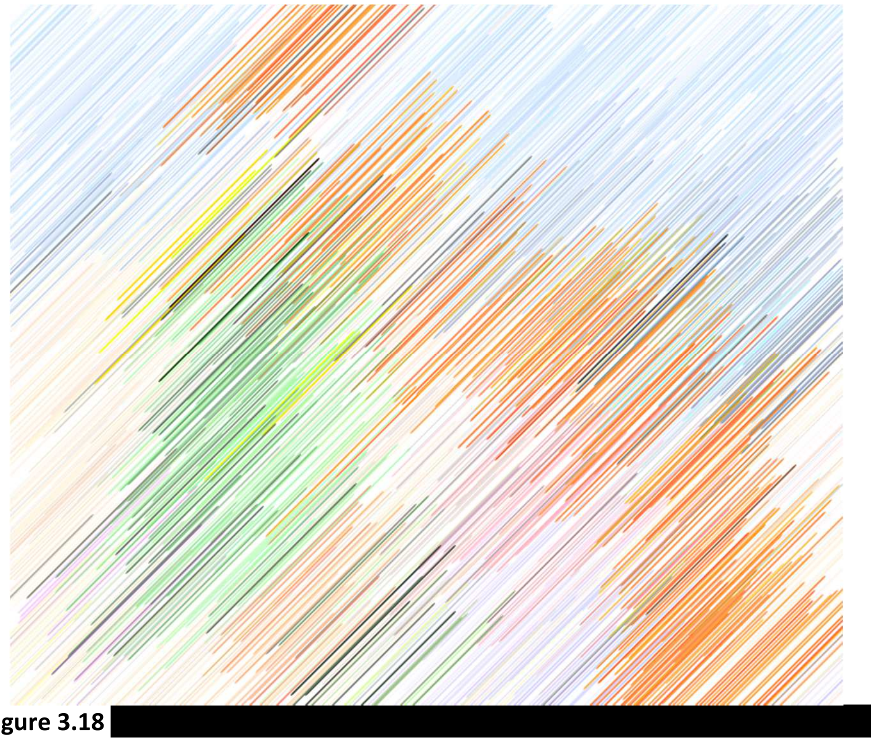


Figure 3.18 [redacted]

3.2.5. Architecture synthesis

The architecture of this robotic cell was defined after all the tests performed. It is a compact, flexible and collaborative cell. Some degrees of freedom were defined to demonstrate the adaptability of this robotic cell to the environment and the conditions where it will be installed. The final architecture is shown in Figure 3.19.

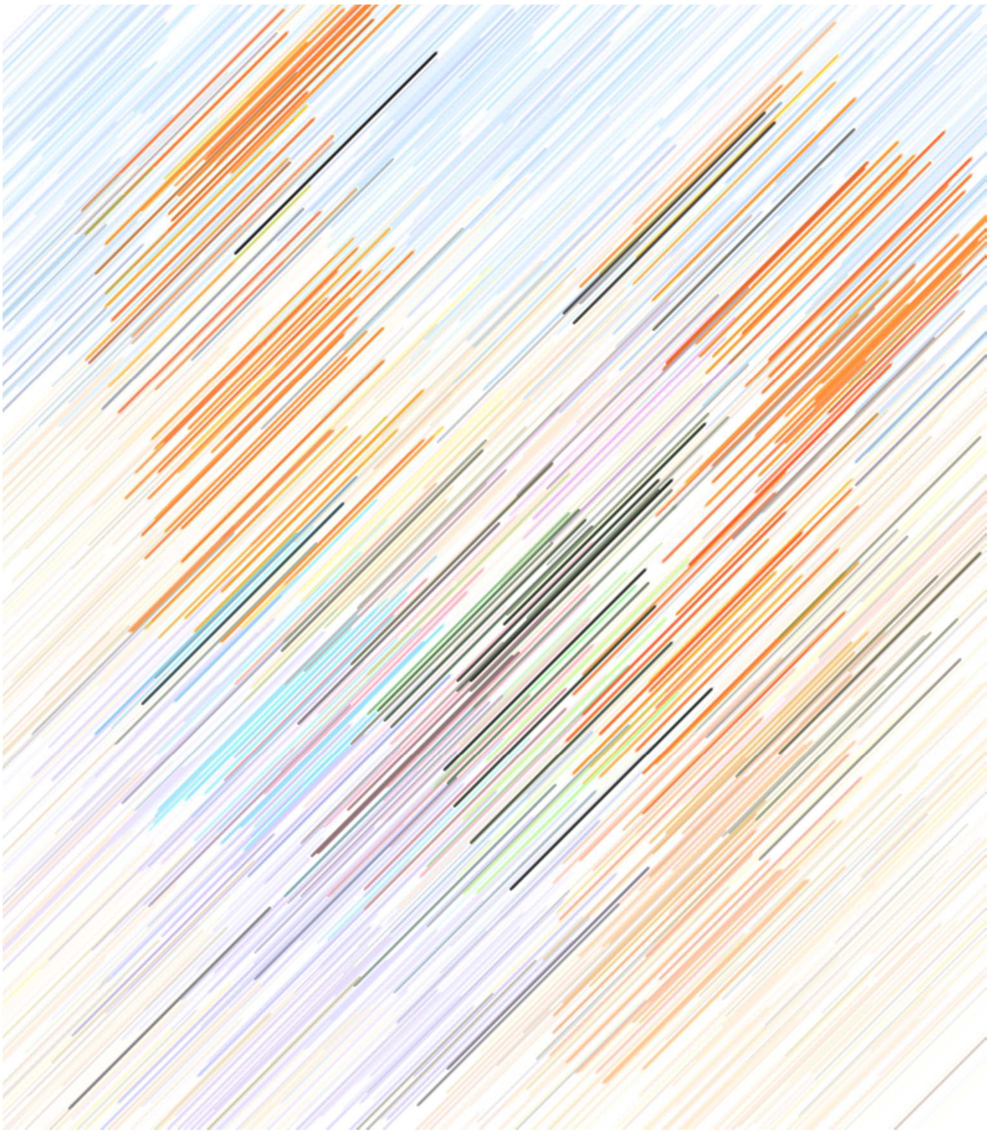


Figure 3.19 Robotic cell architecture

4. CALIBRATION OF A ROBOTIC CELL

A new calibration procedure was created in order to obtain the exact position of different objects in a work cell using a single point laser measurement sensor. This task is very important to verify the position of the objects in relation to the robots or the other components of the work cell. The calibration is even more important when there is an automated system because the paths performed by the robots must be adapted to be coherent with the positions and orientations of the objects obtained.

This work presents a methodology to calibrate a single point laser measurement sensor and also to obtain the positions and orientations of the components composing a robotic cell.

The sensor used is the *"optoNCDT 1700LL-50"* from *Micro-Epsilon*™ (see Figure 4.1) and it has a short range (50 mm). The measurement range begins 45 mm far from the sensor surface where the laser is emitted, which means that 95 mm is the maximum distance that the sensor can measure counting from the sensor surface. This sensor is prepared to be used in the robotics field as it has a long flexible cable. In addition to this, to fix the sensor to the robot flange it is only needed to have a simple support with 3 holes to attach the sensor and the necessary adaption to the robot flange. This support is also important for the calibration of the sensor and to define the Laser frame located at the beginning of the measurement range. The communication protocol to deal with this laser sensor can be find in Annex A.

Figure 4.1 Single point laser measurement sensor

Before starting the measurement with the sensor, it is necessary to calibrate it and well define the transformation between the robot flange and the beginning of the measurement range. This operation is also important to confirm the precision of the sensor measurements.

[redacted]

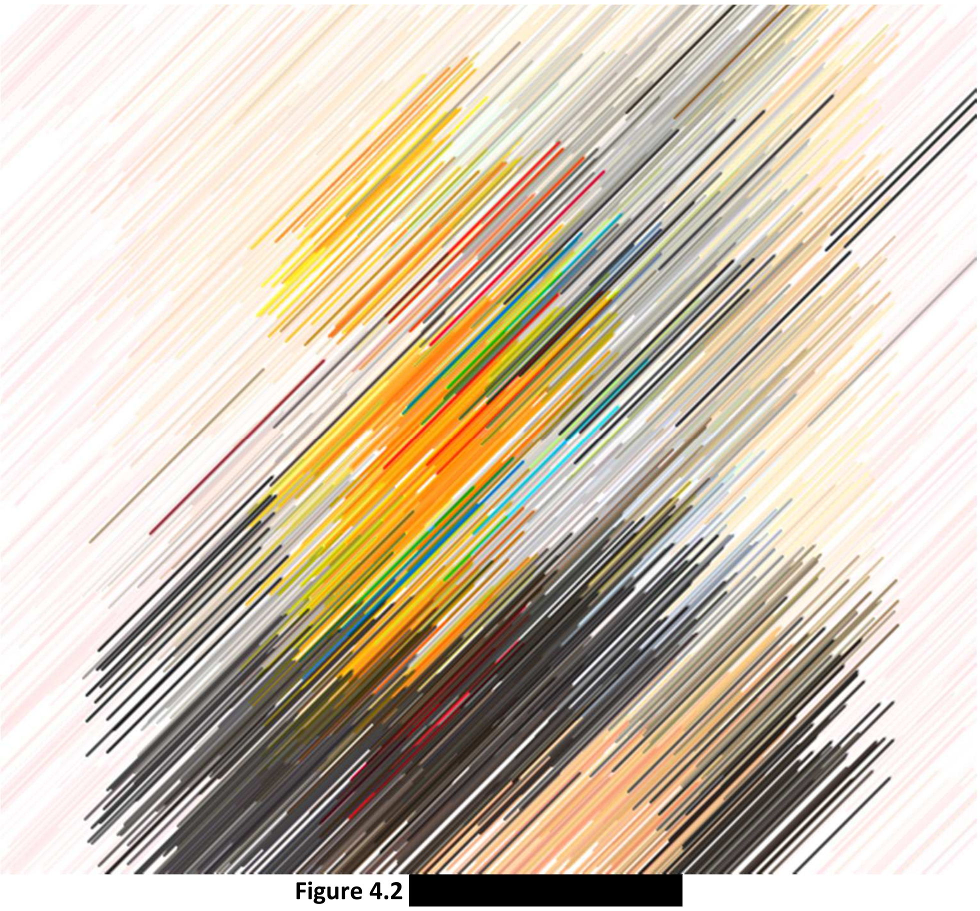


Figure 4.2 [redacted]

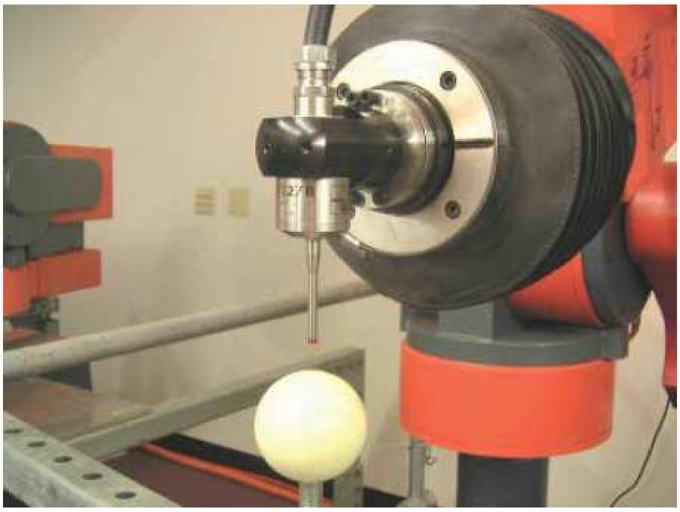


Figure 4.3 Moving trigger calibration [37]

When the laser is well calibrated and the Laser frame is well defined in the robot frame, everything is set to measure with the sensor.

[redacted]

[redacted]

[redacted] [redacted]

[redacted]

[redacted]

[redacted]

[redacted] [redacted]

[redacted]

4.1. Work-object calibration

In order to obtain the position of an object, [redacted]

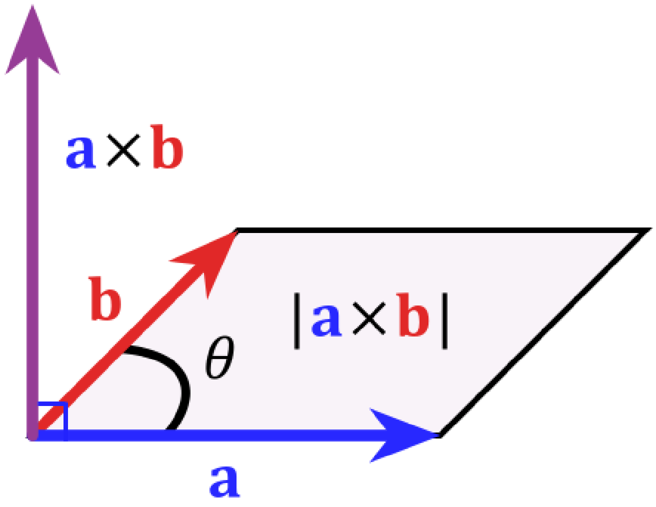


Figure 4.4 Cross Product definition

[redacted]

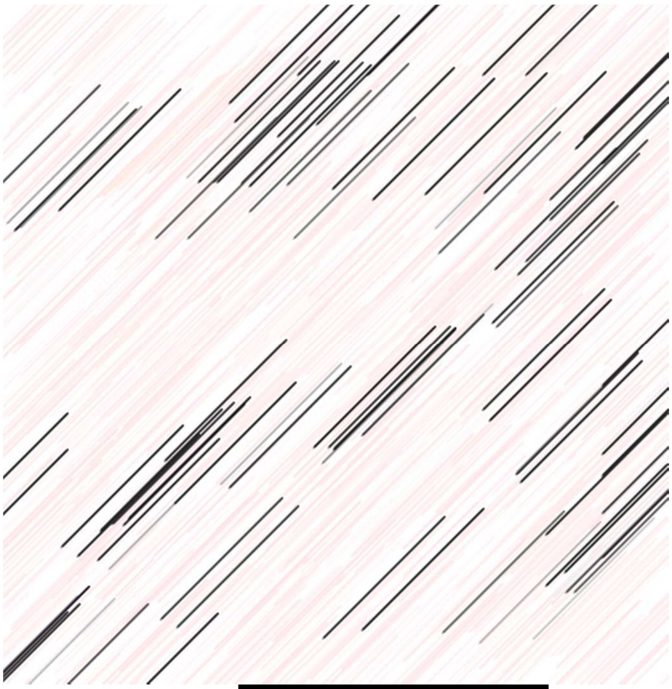


Figure 4.5 [redacted]

[redacted]

[redacted] [redacted]

[redacted] [redacted]

[redacted]

[redacted]

[redacted] [redacted]

[redacted] [redacted]

[redacted] [redacted]

[redacted]

[redacted]

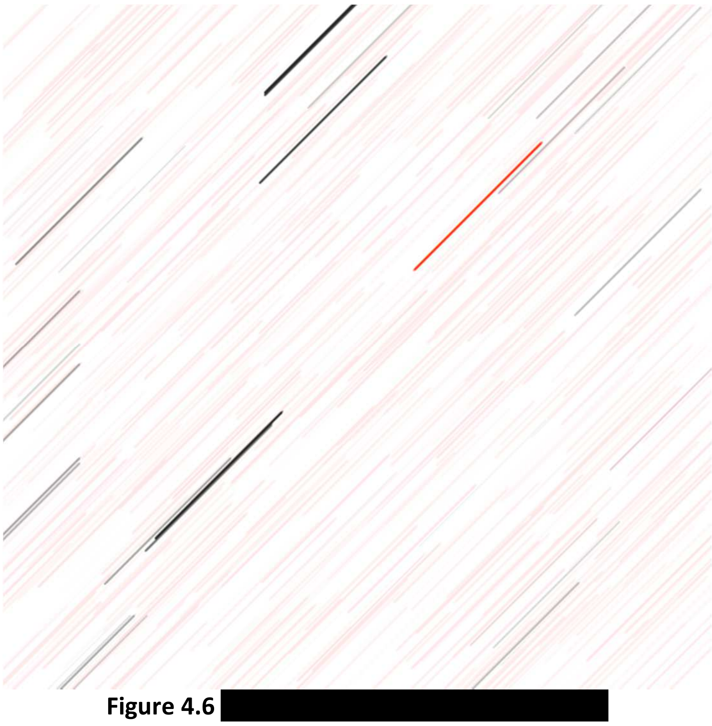


Figure 4.6 [redacted]

[redacted]

[redacted]

[redacted] [redacted]

[redacted] [redacted]

[redacted]

[redacted]

The methodology presented above will be applied in the robotic cell which was designed before. It will be used to define the position and orientation of all the objects in the robotic cell[redacted]

4.2. Cylinder calibration

[redacted] Using this method will allow to obtain the following parameters of the cylinder:

- A point (x_0, y_0, z_0) on its axis;
- A vector (a, b, c) pointing along its axis;
- Its radius r.

Considering that x_i, y_i, z_i are any point on the cylinder surface, the following equation represents the cylindrical equation in the 3D space:

$$X^2 + Y^2 + Z^2 = R^2 \tag{25}$$

Where:

$$X = c(y_i - y_0) - b(z_i - z_0) \tag{26}$$

$$Y = a(z_i - z_0) - c(x_i - x_0) \tag{27}$$

$$Z = b(x_i - x_0) - a(y_i - y_0) \tag{28}$$

$$R = radius \tag{29}$$

Arranging this equation it is possible to obtain the following equation:

$$Ax^2 + By^2 + Cz^2 + Dxy + Exz + Fyz + Gx + Hy + Iz + J = 0 \tag{30}$$

Where:

$$A = (b^2 + c^2) \tag{31}$$

$$B = (a^2 + c^2) \tag{32}$$

$$C = (a^2 + b^2) \tag{33}$$

$$D = -2ab \tag{34}$$

$$E = -2ac \tag{35}$$

$$F = -2bc \tag{36}$$

$$G = -2(b^2 + c^2)x_0 + 2aby_0 + 2acz_0 \tag{37}$$

$$H = 2abx_0 - 2(a^2 + c^2)x_0 + 2bcz_0 \tag{38}$$

$$I = 2acx_0 + 2bcy_0 - 2(a^2 + b^2)z_0 \tag{39}$$

$$\begin{aligned} J = (b^2 + c^2){x_0}^2 + (a^2 + c^2){y_0}^2 + (a^2 + b^2){z_0}^2 - 2bcy_0z_0 - 2acz_0x_0 \\ - 2abx_0y_0 - R^2 \end{aligned} \tag{40}$$

[redacted]

[redacted] [redacted]

[redacted]

[redacted] [redacted]

[redacted]

[redacted]

[redacted] [redacted]

[redacted]

[redacted] [redacted]

[redacted]

[redacted]

[redacted] [redacted]

[redacted]

[redacted] [redacted]

[redacted]

[redacted] [redacted]

[redacted]

[redacted] [redacted]

[redacted]

[redacted] [redacted]

[redacted]

This method of calibrating a cylinder will be applied to the robotic cell. [redacted]

5. CONCLUSIONS

5.1. Synthesis of the work realized

The work presented in this Master thesis consisted on the design of a robotic cell and on the development of its calibration. The need of calibration is due to the fact that this robotic cell is compact and was entirely designed by offline programming.

Along the Chapter 2, a complete Literature Review was made in order to get to know the works performed around the subject of this Master thesis. This research work began by searching for papers about robot calibration but was extended to obtain the relevant works on robotic cell calibration. This allowed to get some knowledge about the necessary procedures to apply in this work.

After getting all the information about the works made around the subject of this thesis, the design of the robotic cell was entirely developed. In Chapter 3, it is described how the architecture of this new robotic cell was created. There it is possible to find from the information known at the beginning of the creation of the robotic cell until the final architecture. All the aspects that were relevant to decide the positioning of the components inside the robotic cell were exposed. Finally, some tests were performed to analyze the degrees of freedom of some components inside the robotic cell. This means that different positions of some components were tested in order to also validate the feasibility of the system. With this information it was possible to define the limits of displacement of some components of the robotic cell.

With all the scenario created, it was time to develop a calibration system to be applied in this robotic cell. This is a fundamental procedure to have the position and orientation of the different components of the robotic cell in one single frame (robot base frame). This calibration procedure was performed with a single point laser measurement sensor. To define the positioning of the components that are supposed to be carried by the robots, a method was developed using the cross product definition. [redacted] This will allow to know the location of the work-object on the robot base frame. Moreover, a method to define a cylindrical form was implemented. [redacted]

To sum up, the complete design of a new robotic cell was created using a numerical simulation software (V-REP) and the calibration procedure to be applied to this cell was also developed.

5.2. Outlooks

This work was performed to a project with an enterprise and the following step is to build this robotic cell in reality. All the simulations performed during the design work will be used to program the robots of this robotic cell.

The calibration procedure must also be tested and adapted to use a Laser Scan measurement sensor. This Laser scan is more powerful than a single point measurement sensor. It will be more efficient to the cylindrical calibration as it gives a set of points (profiles), which will reduce the errors of calculation of the cylinder location.

Figure 5.1 Laser scan measurement sensor

6. BIBLIOGRAPHY

[1] C. Gong, J. Yuan and J. Ni, “Nongeometric error identification and compensation for robotic system by inverse calibration,” *International Journal of Machine Tools & Manufacture 40,* pp. 2119-2137, 2000.

[2] A. Elatta, L. P. Gen, F. L. Zhi, Y. Daoyuan and L. Fei, “An Overview of Robot Calibration,” *Information Technology Journal 3 (1),* pp. 74-78, 2004.

[3] W. Khalil and E. Dombre, Modeling, Identification & Control of Robots, Butterworth Heinemann, 2004.

[4] A. Olabi, M. Damak, R. Béarée, O. Gibaru and S. Leleu, “Improving the accuracy of industrial robots by offline compensation of joints errors,” in *IEEE International Conference on Industrial Technology*, Island of Kos, Greece, 2012.

[5] A. Olabi, “Amélioration de la précision des robots industriels pour des applications d'usinage a grande vitesse,” Lille, France, 2011.

[6] C. Rocha, C. Tonetto and A. Dias, “A comparison between the Denavit-Hartenberg and the screw-based methods used in kinematic modeling of robot manipulators,” *Robotics and Computer-Integrated Manufacturing,* vol. 27, pp. 723-728, 2011.

[7] J. Denavit and R. Hartenberg, “A kinematic notation for lower-pair mechanisms based on matrices,” *Transactions ASME, Journal of Applied Mechanics,* vol. 22(1), pp. 215-221, 1955.

[8] C. Gang, L. Tong, C. Ming, J.-Q. Xuan and S.-H. Xu, “Review on Kinematics Calibration Technology of Serial Robots,” *International Journal of Precision Engineering and Manufacturing,* vol. 15, no. 8, pp. 1759-1774, 2014.

[9] S. Hayati, “Robot arm geometric link parameter estimation,” in *22nd IEEE Conference on Decision and Control,* 1983.

[10] G. Chen, H. Wang and Z. Lin, “Determination of the Identifiable Parameters in Robot Calibration Based on the POE Formula,” *IEEE Transactions on Robotics,* vol. 30, no. 5, 2014.

[11] X. Yang, L. Wu, J. Li and K. Chen, “A minimal kinematic model for serial robot

calibration using POE formula," *Robotics and Computer-Integrated Manufacturing,* vol. 30, pp. 326-334, 2014.

[12] H. Wang, Q. Sun, W. Liu and X. Lu, "Application of Screw Axis Identification Method for Serial Robot Calibration In The Service Robot," in *25th Chinese Control and Decision Conference (CCDC),* 2013.

[13] H.-N. Nguyen, J. Zhou and H.-J. Kang, "A New Full Pose Measurement Method for Robot Calibration," *Sensors,* vol. 13, pp. 9132-9147, 2013.

[14] A. Nubiola and I. A. Bonev, "Absolute calibration of an ABB IRB 1600 robot using a laser tracker," *Robotics and Computer-Integrated Manufacturing,* vol. 29, pp. 236-245, 2013.

[15] A. Joubair, M. Slamani and I. A. Bonev, "Kinematic calibration of a five-bar planar parallel robot using all working modes," *Robotics and Computer-Integrated Manufacturing,* vol. 29, pp. 15-25, 2013.

[16] M. Švaco, B. Šekoranja, F. Šuligoj and B. Jerbić, "Calibration of an Industrial Robot using a Stereo Vision System," *Procedia Engineering,* vol. 69, pp. 459-463, 2014.

[17] Y. Guo, S. Yin, Y. Ren, J. Zhu, S. Yang and S. Ye, "A multilevel calibration technique for an industrial robot with parallelogram mechanism," *Precision Engineering*, vol. 40, pp. 261-272, 2015.

[18] A. Joubair, M. Slamani and I. A. Bonev, "A novel XY-Theta precision table and a geometri procedure for its kinematic calibration," *Robotics and Computer-Integrated Manufacturing,* vol. 28, pp. 57-65, 2012.

[19] M. A. Meggiolaro, G. Scriffignano and S. Dubowsky, "Manipulator calibration using a single endpoint contact constraint," in *26th Biennial Mechanisms and Robotics of the 2000 ASME Design Engineering Technical Conferences*, 2000.

[20] A. Joubair and I. A. Bonev, "Non-kinematic calibration of a six-axis serial robot using planar constraints," *Precision Engineering,* vol. 40, pp. 325-333, 2014.

[21] S. Besnard, W. Khalil and G. Garcia, "Geometric calibration of robots using multiple plane constraints," *Advances in Robot Kinematics,* pp. 61-70, 2000.

[22] H. Hage, P. Bidaud and N. Jardin, "Practical consideration on the identification of the kinematic parameters of the Stäubli TX90 robot," in *13th World Congress in*

Mechanism and Machine Science, 2011.

[23] A. Nubiola, M. Slamani and I. A. Bonev, "A new method for measuring a large set of poses with a single telescoping ballbar," *Precision Engineering*, vol. 37, pp. 451-460, 2013.

[24] A. Nubiola and I. A. Bonev, "Absolute Robot Calibration with a single Telescoping ballbar," *Precision Engineering*, vol. 38, pp. 472-480, 2014.

[25] Y. Wu, A. Klimchik, S. Caro, B. Furet and A. Pashkevich, "Advanced robot calibration using partial pose measurements," IEEE, 2013.

[26] Y. Wu, A. Klimchik, S. Caro, B. Furet and A. Pashkevich, "Geometric calibration of industrial robots using enhanced partial pose measurements and design of experiments," *Robotics and Computer-Integrated Manufacturing*, vol. 35, pp. 151-168, 2015.

[27] T. Li, K. Sun, Y. Jin and H. Liu, "A novel optimal calibration algorithm on a dexterous 6 DOF serial robot-with the optimization of measurement poses number," in *2011 IEEE International Conference on Robotics and Automation*, Shanghai, 2011.

[28] C.-C. D. Lu and J. D. Hayes, "Kinematic calibration of 6R serial manipulators using relative measurements," in *CCToMM Mechanisms, Machines and Mechatronics (M3) Symposium*, 2013.

[29] K. Radkhah, T. Hemker and O. von Stryk, "A Novel Self-Calibration Method for Industrial Robots Incorporating Geometric and Nongeometric Effects," in *Proceedings of 2008 IEEE International Conference on Mechatronics and Automation*, 2008.

[30] A. Watanabe, S. Sakakibara, K. Ban, M. Yamada, G. Shen and T. Arai, "Autonomous Visual Measurement for Accurate Setting of Workpieces in Robotic Cells," *CIRP Annals - Manufacturing Technology*, vol. 54, pp. 13-18, 2005.

[31] E. Nieves, N. Xi, B. Du and Y. Jia, "A Reflected Laser Line Approach for Industrial Robot Calibration," in *The 2012 IEEE/ASME International Conference on Advanced Intelligent Mechatronics*, Kaohsiung, 2012.

[32] E. Nieves, N. Xi, Y. Jia, C. Martinez and G. Zhang, "Development of a Position Sensitive Device and Control Method for Automated Robot Calibration," in *IEEE*

International Conference on Automation Science and Engineering (CASE), 2013.

[33] E. Nieves and N. Xi, “Robot Control System for Multi-position Alignment Used to Automate an Industrial Robot Calibration Approach,” in *IEEE International Conference on Robotics & Automation (ICRA)*, Hong Kong, 2014.

[34] P. Roçadas and D. R. S. McMaster, “A robot cell calibration algorithm and its use with a 3D measuring system,” in *ISIE*, Guimarães, 1997.

[35] W. Wang, F. Liu and C. Yun, “Calibration method of robot base frame using quaternion form,” *Precision Engineering*, vol. 41, pp. 47-54, 2015.

[36] J.-S. Hu, J.-J. Wang and Y.-J. Chang, “Kinematic Calibration of Manipulator Using Single Laser Pointer,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Vilamoura, Algarve, Portugal, 2012.

[37] X. Li, T. A. Fuhlbrigge, S. Choi and B. Zhang, “Automatic Offline Program Calibration in Robotic Cells,” in *The 4th Annual IEEE Internaitonal Conference on Cyber Technology in Automation, Control and Intelligent Systems*, Hong Kong, 2014.

[38] X. Li and B. Zhang, “Toward General Industrial Robot Cell Calibration,” in *IEEE 5th International Conference on Robotics, Automation and Mechatronics (RAM)*, 2011.

[39] J. Wang, L. Wu, M. Q.-H. Meng and H. Ren, “Towards Simultaneous Coordinate Calibrations for Cooperative Multiple Robots,” in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2014)*, Chicago, 2014.

[40] T. Wei, Z. Yuanfan, Z. Wei and L. Wenhe, “Calibration of robotic drilling systems with a moving rail,” *Chinese Journal of Aeronautics*, vol. 27, pp. 1598-1604, 2014.

[41] J. Ge, H. Gu, Q. Lu and Q. Li, “An Automatic Industrial Robot Cell Calibration Method,” in *ISR ROBOTIK*, Berlin, Germany, 2014.

[42] F. S. Cheng, “The Method of Recovering Robot TCP Positions in Industrial Robot Application Programs,” in *IEEE International Conference on Mechatronics and Automation*, Harbin, China, 2007.

[43] P. Y. Tao, G. Yang and M. Tomizuka, “A Calibration Framework for Industrial Robotic Work Cells,” in *IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, Wollongong, 2013.

7. ANNEX A

7.1. Communication protocol to use the single point laser measurement sensor with a Stäubli TX90 robot

A single point laser measurement sensor is operated by a computer and it should use its own software. However, when it is needed to communicate with a robot, this software does not support the robot language. In addition to this, each brand of robots has its own language of programming too. To overtake these issues, a program was developed using a standard language (C++) to be possible to make the robot and the sensor communicate between them. This is very important to automate the system of calibration. This will allow to program various movements of the robot and program signals ordering the laser sensor to take measures. This communication has 4 devices shown in Figure 7.1: external computer, laser sensor, robot and its controller.

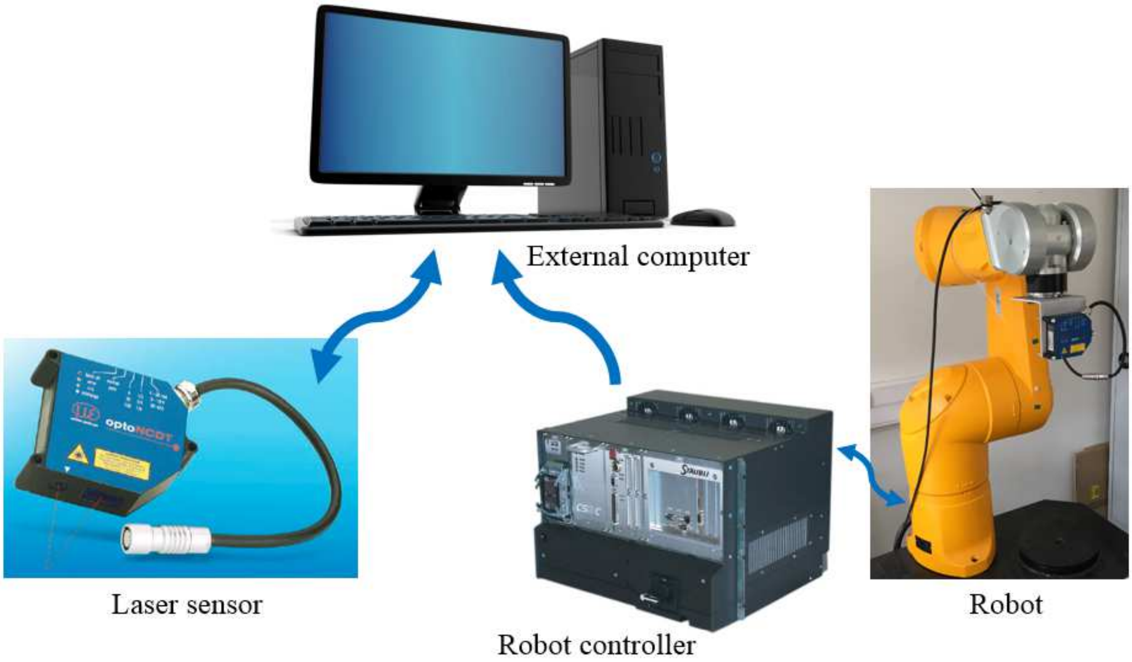
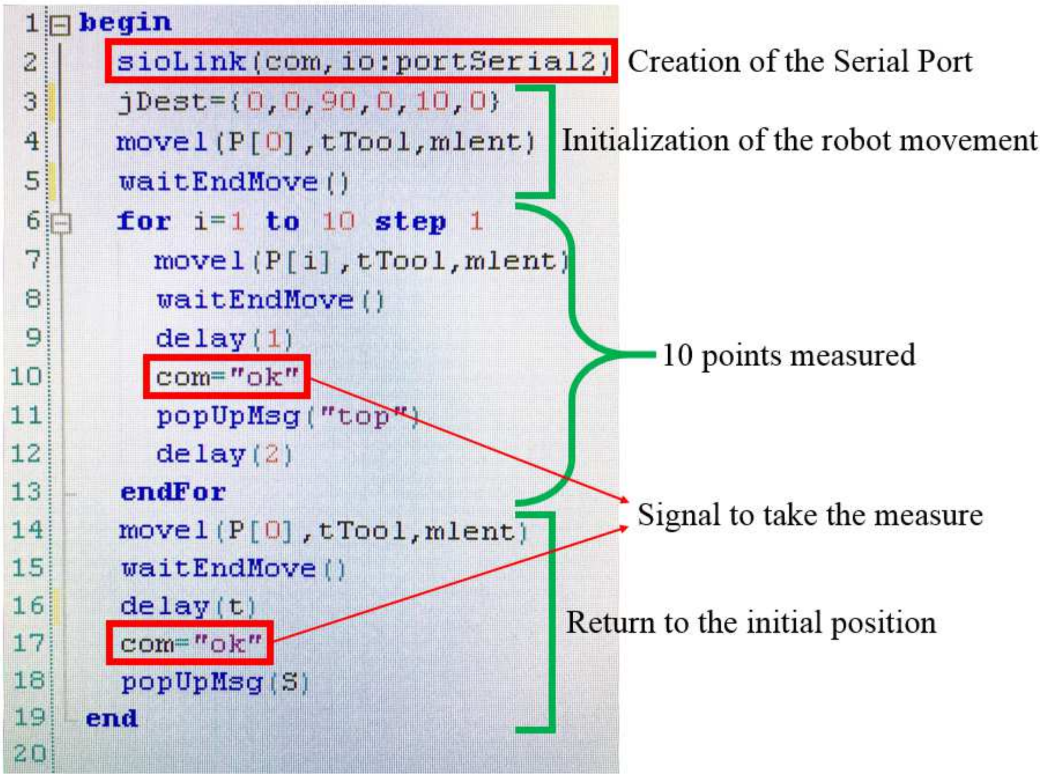


Figure 7.1 Communication scheme

These devices are all linked as shown in the figure above. The connection of the laser sensor is by a RS-232 serial interface (USB cable). RS-232 is the most common serial interface and ships as a standard component on most Windows-compatible desktop computers. RS-232 only allows for one transmitter and one receiver on each line. It is only necessary to install the driver of the sensor and establish the connection with the correct COM Port of the computer. The robot controller and the computer are connected by a Serial Port. A Serial Port is a serial communication physical interface through which information transfers in or out one bit at a time.

To automate all the process of taking measures with the laser sensor it is necessary to create two programmes in different languages. One is programmed to control the robot movements and give the order to take the measures in the robot language (VAL3 for Stäubli). This programme will run on the robot controller (see Figure 7.2). The other one is a programme created with Microsoft Visual Studio™ in C++ language. This second programme will run on the external computer and will be reading continuously the Serial Port that is connected with the robot controller. The program will be waiting for the signals to take measures and will save these measures in a file that will then be used to make the necessary calculations.



```
begin
  sioLink(com,io:portSerial2)
  jDest={0,0,90,0,10,0}
  movel(P[0],tTool,mlent)
  waitEndMove()
  for i=1 to 10 step 1
    movel(P[i],tTool,mlent)
    waitEndMove()
    delay(1)
    com="ok"
    popUpMsg("top")
    delay(2)
  endFor
  movel(P[0],tTool,mlent)
  waitEndMove()
  delay(t)
  com="ok"
  popUpMsg(S)
end
```

Figure 7.2 VAL3 script to command the Stäubli TX90 Robot

In VAL3 it is necessary to define the Serial Port. Firstly, the Serial Port must be created in the background program of the inputs and outputs of the robot (see Figure

7.3). Secondly, the socket “com” must be created in the “sio” (inputs and outputs) data of the main program (see Figure 7.4).

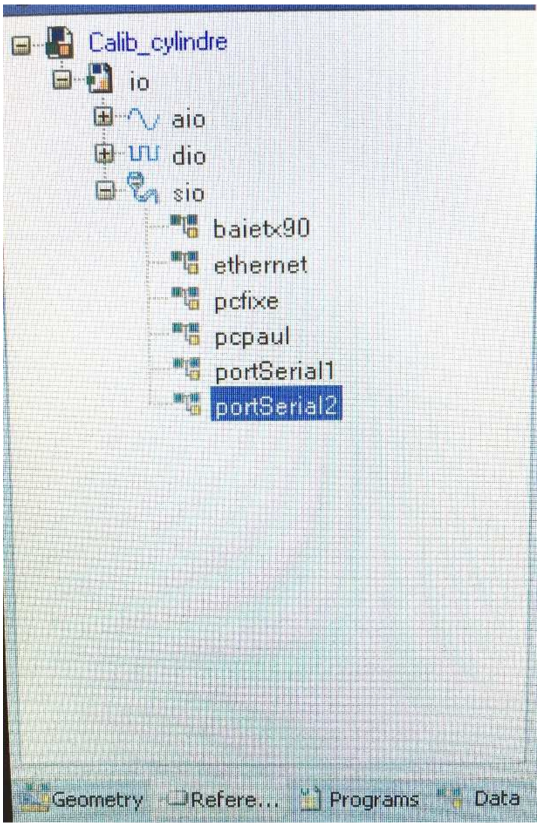


Figure 7.3 Creation of the Serial Port

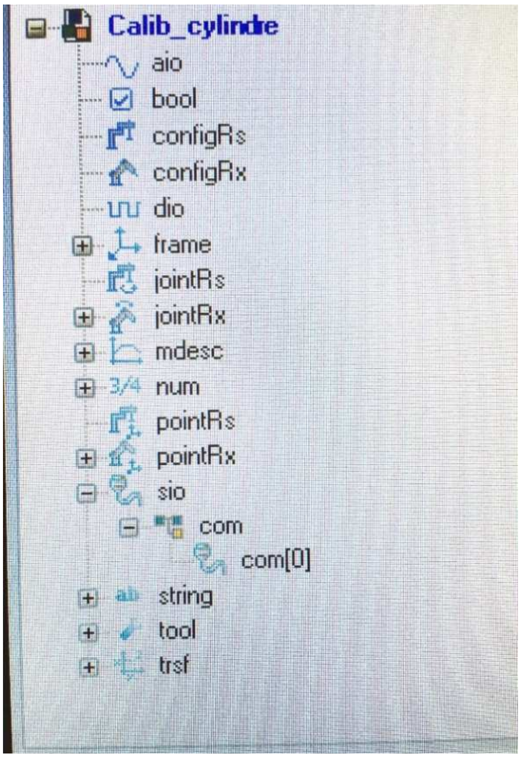
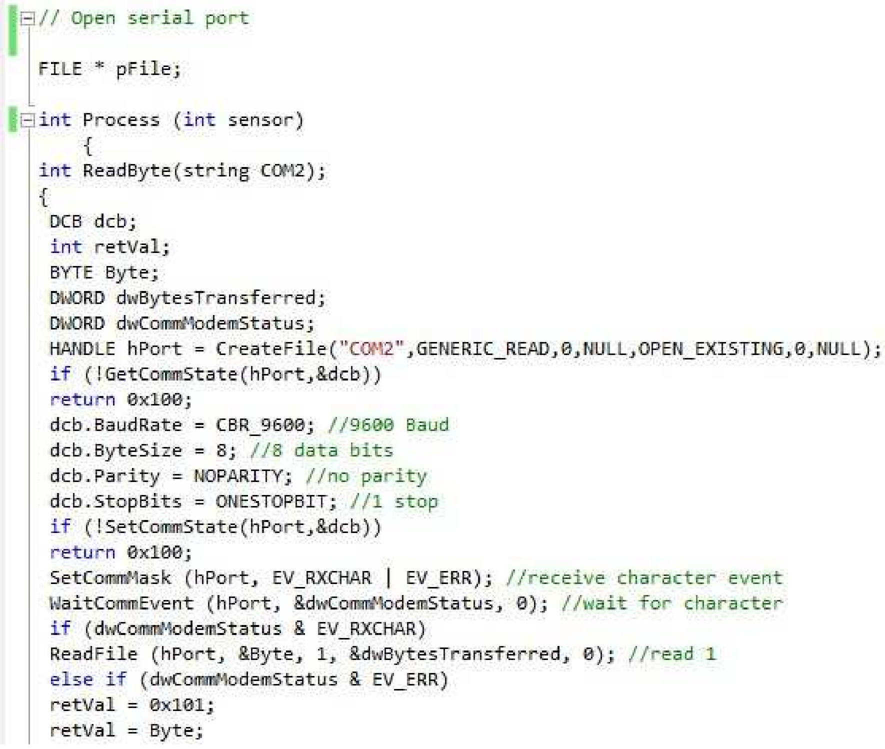


Figure 7.4 Socket creation

As said before, a program will be running on the external computer to save a file with the measures taken. In Figure 7.5 it is shown the Serial Port connection and the continuous reading of the data in the Serial Port.



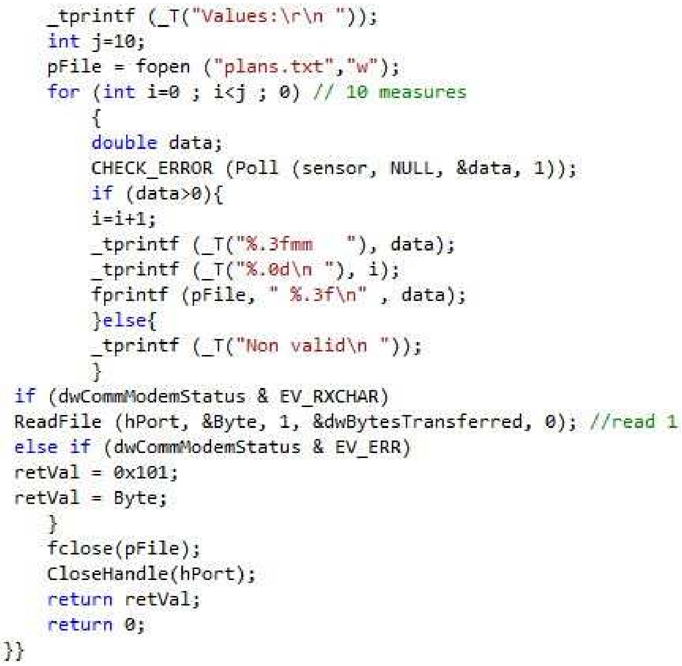
```
// Open serial port

FILE * pFile;

int Process (int sensor)
    {
int ReadByte(string COM2);
{
 DCB dcb;
 int retVal;
 BYTE Byte;
 DWORD dwBytesTransferred;
 DWORD dwCommModemStatus;
 HANDLE hPort = CreateFile("COM2",GENERIC_READ,0,NULL,OPEN_EXISTING,0,NULL);
 if (!GetCommState(hPort,&dcb))
 return 0x100;
 dcb.BaudRate = CBR_9600; //9600 Baud
 dcb.ByteSize = 8; //8 data bits
 dcb.Parity = NOPARITY; //no parity
 dcb.StopBits = ONESTOPBIT; //1 stop
 if (!SetCommState(hPort,&dcb))
 return 0x100;
 SetCommMask (hPort, EV_RXCHAR | EV_ERR); //receive character event
 WaitCommEvent (hPort, &dwCommModemStatus, 0); //wait for character
 if (dwCommModemStatus & EV_RXCHAR)
 ReadFile (hPort, &Byte, 1, &dwBytesTransferred, 0); //read 1
 else if (dwCommModemStatus & EV_ERR)
 retVal = 0x101;
 retVal = Byte;
```

Figure 7.5 Declaration of the Serial Port

In Figure 7.6 it is shown the "for" cycle that is ready to take the number of valid measures needed. In this example, 10 measures are taken.



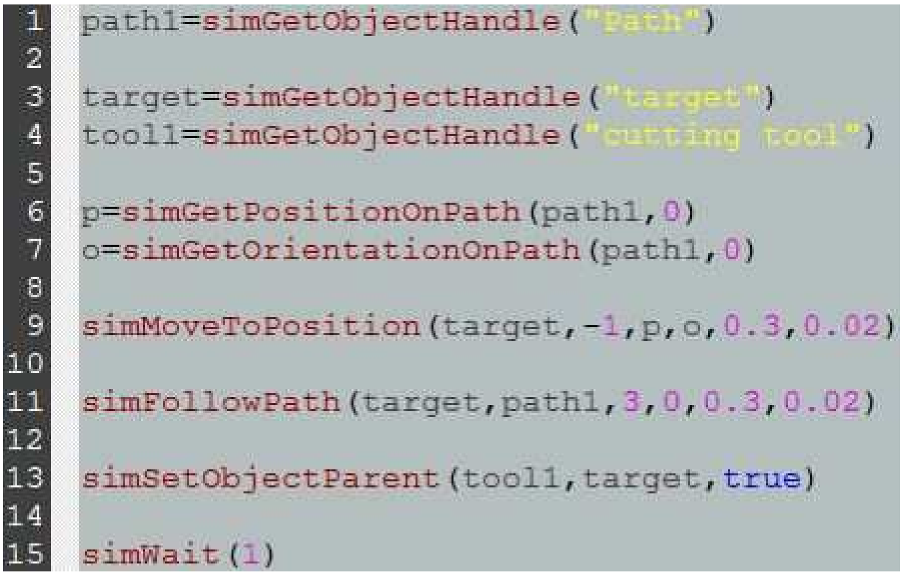
```
     _tprintf (_T("Values:\r\n "));
     int j=10;
     pFile = fopen ("plans.txt","w");
     for (int i=0 ; i<j ; 0) // 10 measures
         {
         double data;
         CHECK_ERROR (Poll (sensor, NULL, &data, 1));
         if (data>0){
         i=i+1;
         _tprintf (_T("%.3fmm   "), data);
         _tprintf (_T("%.0d\n "), i);
         fprintf (pFile, " %.3f\n" , data);
         }else{
         _tprintf (_T("Non valid\n "));
         }
  if (dwCommModemStatus & EV_RXCHAR)
  ReadFile (hPort, &Byte, 1, &dwBytesTransferred, 0); //read 1
  else if (dwCommModemStatus & EV_ERR)
  retVal = 0x101;
  retVal = Byte;
     }
     fclose(pFile);
     CloseHandle(hPort);
     return retVal;
     return 0;
}}
```

Figure 7.6 Programation of the measures

8. ANNEX B

8.1. V-REP script to program robot movements

The numerical simulation software (V-REP) uses a specific language of scripting (LUA). The child scripts in V-REP are very useful to program robot movements. In Figure 8.1 there is an example of the functions used on this work.



```
path1=simGetObjectHandle("Path")

target=simGetObjectHandle("target")
tool1=simGetObjectHandle("cutting tool")

p=simGetPositionOnPath(path1,0)
o=simGetOrientationOnPath(path1,0)

simMoveToPosition(target,-1,p,o,0.3,0.02)

simFollowPath(target,path1,3,0,0.3,0.02)

simSetObjectParent(tool1,target,true)

simWait(1)
```

Figure 8.1 V-REP script (LUA)

Where:

- *simGetObjectHandle*: is the function to declare any object, previously added to the scene, on the LUA script;
- *simGetPositionOnPath*: it is used to obtain the initial position on a previously created path on the scene;
- *simGetOrientationOnPath*: it is used to obtain the initial orientation on a previously created path on the scene;
- *simMoveToPosition*: function to make the robot move to a certain position;
- *simFollowPath*: this function makes the robot follow a pre-established path;
- *simSetObjectParent*: this function is used to attach a tool to the robot to be carried;
- *simWait*: this is a function that pauses the simulation.