2
Kinematic Modeling for Calibration of Manipulators

2.1.
Introduction

Kinematic modeling is the first step in the calibration process of the manipulator. The kinematic model calculates the movement of the end-effector of the manipulator from the movements of the joints. The calibration process includes alteration of the kinematic model to compensate for the estimated errors. The errors can be classified as either static or dynamic errors. The static errors can be either repetitive or random. The random errors can be caused by backlash and friction in the moving joints of the manipulator. The repetitive errors are caused by errors in the construction process, causing a deviation from the nominal measurements of the manipulator. In this thesis only the static errors will be estimated since the random errors are difficult to model and are expected to be small.

Homogeneous transformations represent a suitable way to describe translations and rotations simultaneously. The different reference frames (coordinate systems) in the respective joints of the manipulator TA-40 studied in this work, are shown in Figure 2. The relative translation and orientation of every joint can be described by a 4x4 homogeneous matrix. Eventually, the coordinates of the end effector relative to the manipulator base can be calculated by multiplying all the transformation matrices representing the respective joints.

This chapter is organized in the following way: Section 2.2 describes the basic concepts of kinematics necessary to model a manipulator. Section 2.3 describes the Denavit-Hartenberg notation (DH) that is used to model the TA-40. Section 2.4 describes the techniques used to calibrate the manipulator. Section 2.5 deduces the inverse kinematics for the manipulator.

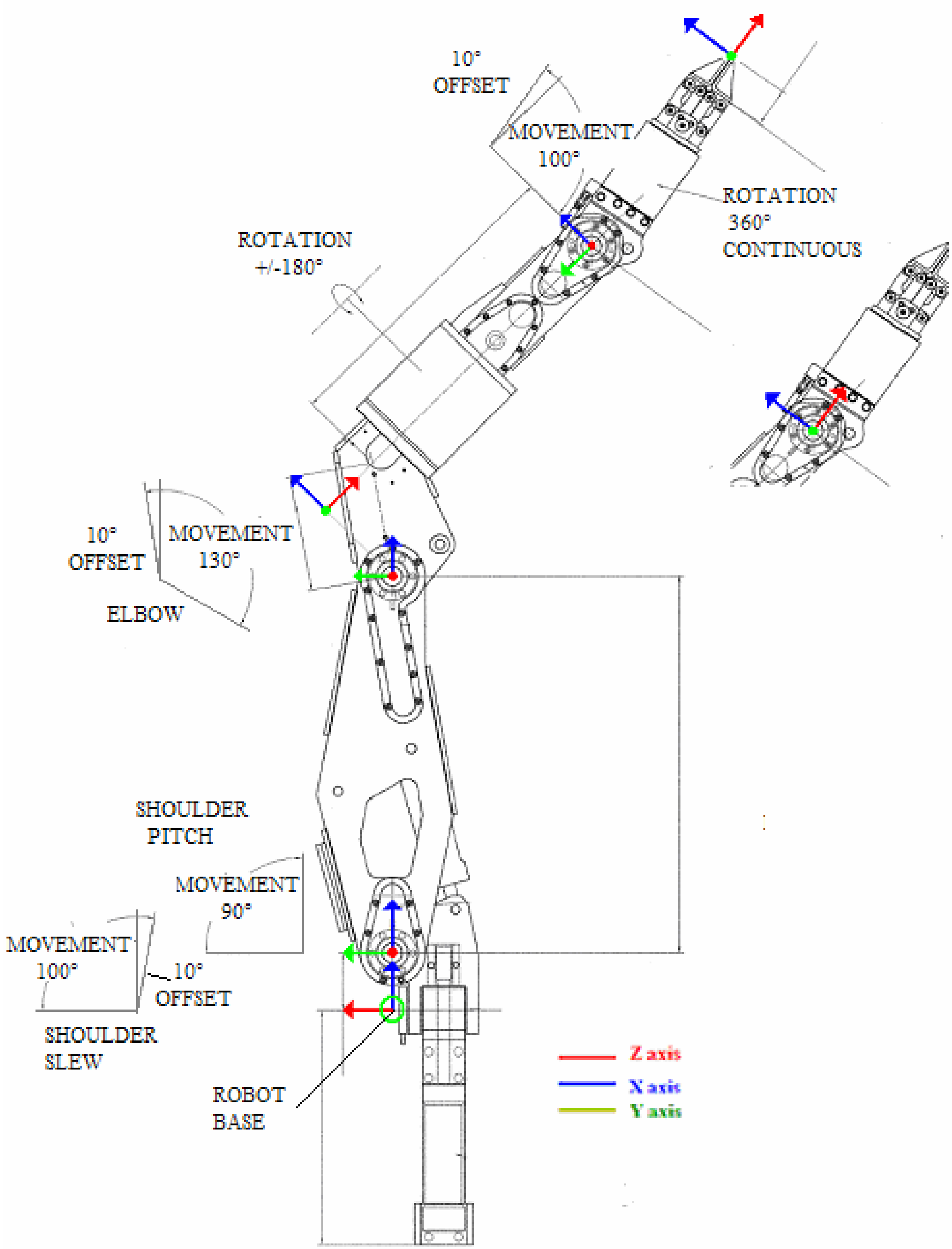

Figure 2 - Coordinate systems of the manipulator

2.2. Basic Concepts of Kinematics

The movement of a rigid body in space can be fully described by a translation, *t*, together with a rotation matrix, R. Together, the translation and rotation form a 4 x 4 homogeneous transformation matrix, *T*.

$$T = \begin{bmatrix} & R & & t \\ & & & \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (1)$$

where the rotation matrix, *R*, is an orthonormal three dimensional matrix. The translation, *t*, is a three dimensional vector.

The movement can be divided into three steps. The movement relative to one axis is described by the rotation angle (θ_x, θ_y *or* θ_z) around the axis together with translation (t_x, t_y *or* t_z) parallel to the same axis. A movement in space can then be described by multiplying the three matrices representing the three axes x, y and z.

$$T(t_x,t_y,t_z,\theta_x,\theta_y,\theta_z) = T(t_z,\theta_z)\cdot T(t_z,\theta_z)\cdot T(t_z,\theta_z) \quad (2)$$

The movement relative to the x axis is described by:

$$T(t_x,\theta_x) = \begin{bmatrix} 1 & 0 & 0 & t_x \\ 0 & \cos\theta_x & -\sin\theta_x & 0 \\ 0 & \sin\theta_x & \cos\theta_x & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

The movement relative to the y axis is described by:

$$T(t_y,\theta_y) = \begin{bmatrix} \cos\theta_y & 0 & \sin\theta_y & 0 \\ 0 & 1 & 0 & t_y \\ -\sin\theta_y & 0 & \cos\theta_y & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (4)$$

The movement relative to the z axis is described by:

$$T(t_z,\theta_z) = \begin{bmatrix} \cos\theta_z & -\sin\theta_z & 0 & 0 \\ \sin\theta_z & \cos\theta_z & 0 & 0 \\ 0 & 0 & 1 & t_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (5)$$

2.3. The Denavit-Hartenberg Convention

The Denavit-Hartenberg notation uses a homogeneous transformation (4 x 4 matrix) to describe the kinematic relationship between a pair of adjacent links. The transformation matrix between the end-effector and the robot base is found by multiplying all the matrices representing the different links and joints. The notation represents a slight simplification of a full 3D movement. It only considers the movements along the x and z axes giving the two adjacent reference frames a common normal along the x axis. The common normals are parallel when $\theta_i = 0°$.

Figure 3 shows a pair of adjacent links, link i-1 and link i, and their associated joints, joint i+1 and joint i. The line $\overline{H_i O_i}$ is called the common normal between the axis i and i+1. The two joints represent two different coordinate systems. In the DH notation, the origin of coordinate system number i is situated in the intersection between axis i+1 and the common normal between the joints i and i+1, as shown in figure 3.

The origin of the i-th coordinate frame O_i is located at the intersection of joints axis i+1 and the common normal between joint axes i and i+1, as shown in figure 3. The frame of link i is at joint i+1 rather than at joint i. The x_i axis is directed along the extension line of the common normal. The z_i axis is along the joint axis i+1. The y_i axis is chosen according to the right-hand rule.

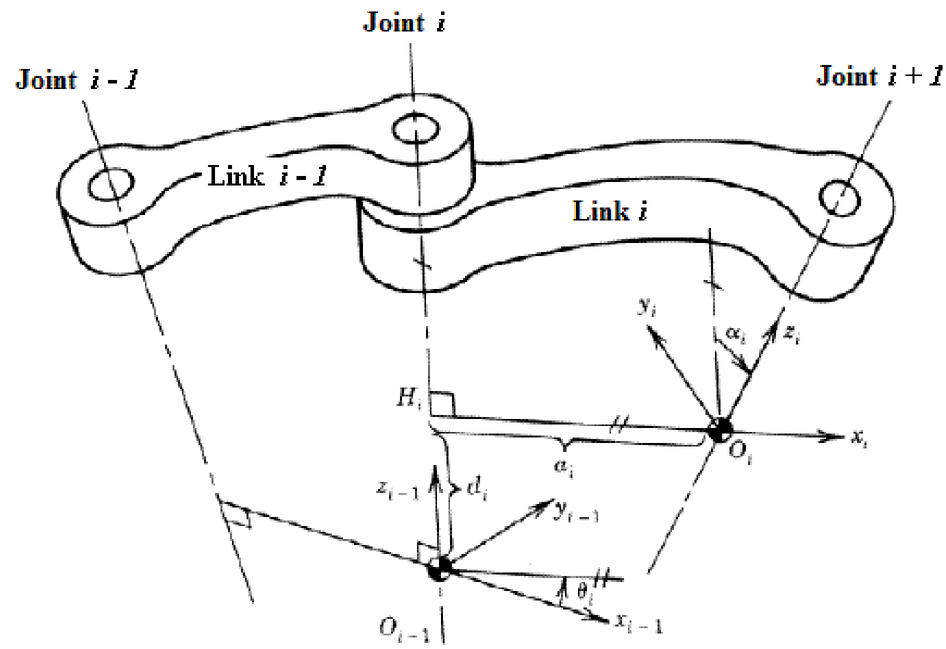

Figure 3 - Denavit-Hartenberg parameters [5]

The relative position of the two frames (coordinate systems) can be completely determined by the following parameters:

a_i – length of the common normal

d_i – distance between the origin O_{i-1} and H

α_i – the angle between the joint axes in the right hand sense.

θ_i – the angle between the x_{i-1} axis and the common normal H_iO_i measured about the z_{i-1} axis in the right hand sense.

For a manipulator with only rotary joints, the parameters a_i, α_i and d_i are constants determined by the geometry of the link, while θ_i varies while the joint rotates. If the joint is prismatic d_i will change according to the movement of the joint.

The DH notation is associated to the coordinate axes $\{x_i,y_i,z_i\}$, i=0,...,n for every link. The base, the end-effector and every link is regarded as a separate coordinate system. This means that a manipulator of n joints will have coordinate systems 0 to n. System 0 represents the base and system n the end-effector.

The transformation matrix from the end-effector to the base coordinates is given by Eq. (6)

$$T_n^0 = A_1^0 A_2^1 A_1^0 \ldots A_i^{i-1} \ldots . A_n^{n-1} \quad (6)$$

where A_i^{i-1} is the homogeneous transformation matrix between coordinate system i-1 relative to coordinate system i.

The transformation of coordinates A_i^{i-1} is therefore represented by its 4 parameters ($\theta_{i,}$ $a_{i,}$ $d_{i,}$ α_i). The construction of the matrices A_i for the links i=1,2,...,n-1 is shown. If the *i*-th joint is revolute, the following transformations are necessary to transform the *i+1* to the system *i*.

The coordinate transformation is done in 4 steps which together constitute the Denavit-Hartenberg transformation matrix:

- Rotate the *(i-1)*-th coordinate system around the x_i axis by an angle α_i
- Translate the coordinate system a distance d_i along the z_{i-1} axis so that the origin of the moving system reaches the intersection between the common normal and normal of the i-th joint.
- Translate the coordinate system a distance a_i along the x_i axis.
- Rotate the coordinate system *i-1* around the axis z_{i-1} an angle θ_i, so that the *x* axis of the moving coordinate system is parallel x_i axis.

The transformation matrix is then found by multiplying the movements along the x axis and the z axis:

$$A_i^{i-1} = T_z \cdot T_x = \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & 0 \\ \sin\theta_i & \cos\theta_i & 0 & 0 \\ 0 & 0 & 1 & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} 1 & 0 & 0 & a_i \\ 0 & \cos\alpha_i & -\sin\alpha_i & 0 \\ 0 & \sin\alpha_i & \cos\alpha_i & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (7)$$

Giving the transformation matrix between each joint:

$$A_i^{i-1} = \begin{bmatrix} \cos\theta_i & -\sin\theta_i\cos\alpha_i & \sin\theta_i\sin\alpha_i & a_i\cos\theta_i \\ \sin\theta_i & \cos\theta_i\cos\alpha_i & -\cos\theta_i\sin\alpha_i & a_i\sin\theta_i \\ 0 & \sin\alpha_i & \cos\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{8}$$

2.4. Classic Manipulator Calibration

By applying the nominal measurements of the manipulator in the DH notation, it is possible to estimate the approximate position and orientation of the manipulator end-effector. In order to increase the precision of the manipulator an error model has to be introduced. The errors are then found by calibration.

The method used will only find the repetitive errors in the robot frame. These errors are caused by precision errors in the fabrication of the manipulator, meaning that the nominal values of the links and joints deviate from the actual ones. These errors do not include play in the joints or other random errors that might occur.

To model the repetitive errors, a notation called generalized errors model will be used. It consists of a homogeneous matrix with 6 parameters. The error matrices are multiplied to the transformation matrices of the DH notation to estimate the effect of the errors in the joints and links.

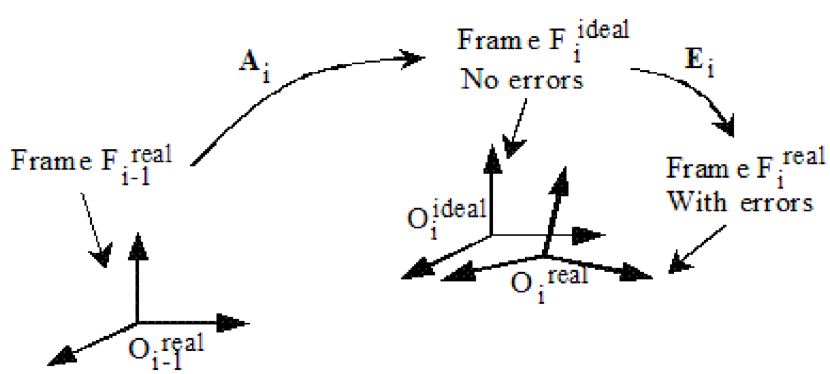

Figure 4 - Translation and rotation with effects of errors in the *i*-th link [6]

The matrix $\boldsymbol{E}_i$ transforms the rotation and translation between the two frames O_i and O_{i-1}. O_i^{real} represents the actual coordinate system of the i-th frame including the errors. O_i^{ideal} represents the ideal position attained by the DH notation using the nominal values of the manipulator. Equation (9) shows the matrix containing the errors. The rotation errors are small, meaning that $\sin(\varepsilon) \approx \varepsilon$ and $\cos(\varepsilon) \approx 1$. This gives:

$$E_i = \begin{bmatrix} 1 & -\varepsilon_r & \varepsilon_s & \varepsilon_x \\ \varepsilon_r & 1 & -\varepsilon_p & \varepsilon_y \\ -\varepsilon_s & \varepsilon_p & 1 & \varepsilon_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{9}$$

where $\varepsilon_{p,i}$, $\varepsilon_{s,i}$ and $\varepsilon_{r,i}$ represent the rotation error around the axes x_i, y_i and z_i respectively. The six parameters , $\varepsilon_{x,i}$, $\varepsilon_{y,i}$, $\varepsilon_{z,i}$, $\varepsilon_{s,i}$, $\varepsilon_{r,i}$, and $\varepsilon_{p,i}$ are called the generalized error parameters. Due to the linearization of the errors, the error matrix is not homogeneous since the rotation matrix is not orthonormal, but in practical terms the matrices can be regarded as homogeneous when the errors are small. For a manipulator with n degrees of freedom there are 6(n+1) generalized errors that can be described as a 6(n+1) x 1 vector: $\varepsilon = [\varepsilon_{x,0}, \ldots, \varepsilon_{x,i}, \varepsilon_{y,i}, \varepsilon_{z,i}, \varepsilon_{s,i}, \varepsilon_{r,i}, \varepsilon_{p,i}, \ldots, \varepsilon_{p,n}]^T$. If the manipulator is calibrated relative to its own base, the error matrix $\boldsymbol{E}_0$ can be eliminated, reducing the number of errors to 6n.

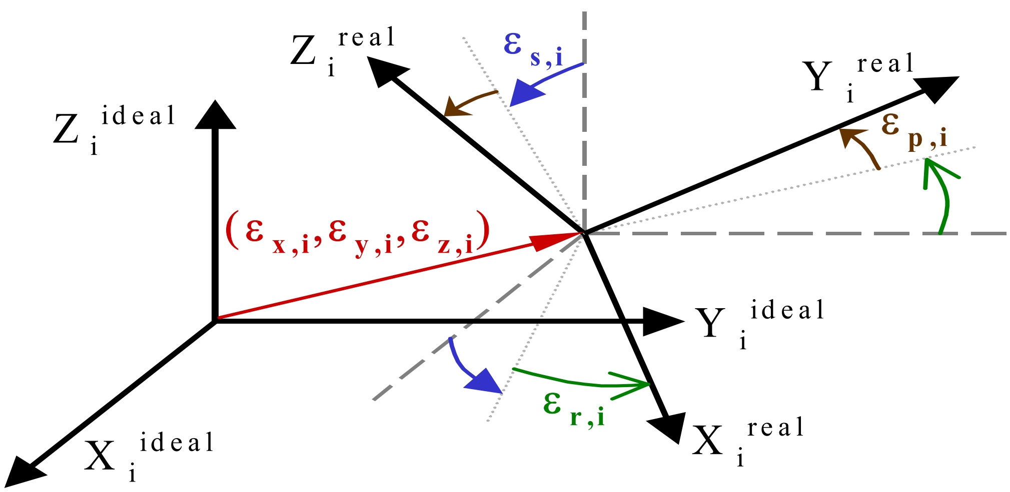

Figure 5 - Generalized errors for the i-th link. $\varepsilon_{p,i}$, $\varepsilon_{s,i}$, $\varepsilon_{r,i}$ represent the rotation around the x,y and z-axes respectively. [6]

By introducing the generalized errors, Eq.(6) can be extended to the form given in Eq.(10).

$$T_n^0(\theta,\varepsilon) = E_0 A_1^0 E_1 A_2^1 E_2 ... A_{n-1}^{n-2} E_{n-1} A_n^{n-1} E_n \qquad (10)$$

The matrix T_n^0 is a 4x4 matrix that describes the orientation and position of the manipulator end-effector relative to its base as a function of the joint parameter vector $\boldsymbol{\theta}=[\theta_1\ \theta_2\ ...\ \theta_n]^T$ and the generalized errors $\boldsymbol{\varepsilon}$.

The pose error of the end-effector of the manipulator, $\boldsymbol{\Delta X}$, denotes the difference between the actual position and the ideal position and can be described by a 6x1vector where three elements represent translation and three elements represent rotation.

$$\Delta\mathbf{X} = \mathbf{X}^{\mathbf{real}} - \mathbf{X}^{\mathbf{ideal}} \qquad (11)$$

Since the generalized errors are small, $\boldsymbol{\Delta X}$ can be estimated by the following equation:

$$\Delta\mathbf{X} = \mathbf{J}_e\, \varepsilon \qquad (12)$$

Where $\boldsymbol{J_e}$ is the Jacobian matrix 6x6(n+1) of the end-effector error $\boldsymbol{\Delta X}$ with respect to the generalized error vector $\boldsymbol{\varepsilon}$, also known as the Identification Jacobian matrix [7]. $\boldsymbol{J_e}$ depends on the system configuration, geometry and task loads. However, only the geometric parameters are treated in this thesis, the manipulator compliance is neglected.

The generalized errors, ε, can be found by a calibration procedure where a number of pose measurements are performed and the generalized errors are estimated by Eq.(12). When the generalized errors are estimated, the errors can be compensated in the control software. A block diagram of the compensation algorithm is shown in Figure 6.

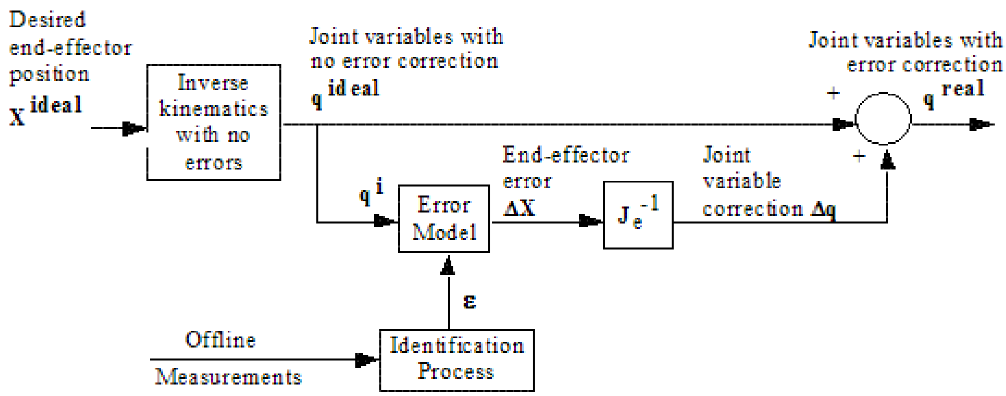

Figure 6 – Error compensation block diagram [6]

This diagram works when the true position can be estimated using the generalized error model. In [8] there is presented a method to correct the trajectory of a robot when the actual position is continuously updated by internal or external sensors.

To estimate the generalized errors, the end-effector pose must be measured in a finite number of different positions. Since it is easier to measure the end-effector position than the orientation, the calibration can be performed by using only the position of the end-effector without measuring its orientation.

The end-effector position has 6 parameters, 3 rotational and 3 translatory. If all the 6 components of ΔX are measured, for a manipulator of n degrees of freedom the generalized errors ε can be calculated measuring ΔX in m different configurations, writing Eq.(12) m times:

$$\Delta\mathbf{X}_t = \begin{bmatrix} \Delta\mathbf{X}_1 \\ \Delta\mathbf{X}_2 \\ \dots \\ \Delta\mathbf{X}_m \end{bmatrix} = \begin{bmatrix} \mathbf{J}_e(\theta_1) \\ \mathbf{J}_e(\theta_2) \\ \dots \\ \mathbf{J}_e(\theta_m) \end{bmatrix} \cdot \boldsymbol{\varepsilon} = \mathbf{J}_t \cdot \boldsymbol{\varepsilon} \qquad (13)$$

In Eq.(13), ΔX_t is a m x 1 vector formed by all the measurements of ΔX in m different configurations and J_t is the $6m$ x $6(n+1)$ matrix formed by m Identification Jacobians, called the Total Identification matrix. It is desirable that the number of measurements m is much higher than n in order to reduce noise effects.

When the errors are constant, the least square estimate of ε can be calculated by Eq.(14).

$$\hat{\varepsilon} = \left(\mathbf{J}_t^T \mathbf{J}_t\right)^{-1} \mathbf{J}_t^T \cdot \Delta \mathbf{X}_t \tag{14}$$

Unfortunately, the Identification Jacobian matrix usually contains columns that are linearly dependent. Therefore Eq. (14) will achieve poor accuracy due to poor matrix conditioning [7]. This occurs when some of the generalized errors are redundant, meaning that it is not possible to distinguish each error's influence on the end-effector position.

2.5. Elimination of Redundant Errors

In order to solve Eq.(14), all the linearly dependent errors have to be eliminated. Here a detailed explanation of the elimination procedure will be given.

The columns , $\boldsymbol{J}_{x,i}$, $\boldsymbol{J}_{y,i}$, $\boldsymbol{J}_{z,i}$, $\boldsymbol{J}_{s,i}$, $\boldsymbol{J}_{r,i}$ and $\boldsymbol{J}_{p,i}$, of $\boldsymbol{J}_e$ are associated to each of the generalized errors $\varepsilon_{x,i}$, $\varepsilon_{y,i}$, $\varepsilon_{z,i}$, $\varepsilon_{s,i}$, $\varepsilon_{r,i}$ and $\varepsilon_{p,i}$, respectively. Equation (12) can be written in the form:

$$\Delta X = \begin{bmatrix} J_{x,0} & \dots & J_{x,i} & J_{y,i} & J_{z,i} & J_{s,i} & J_{r,i} & J_{p,i} & \dots & J_{p,n} \end{bmatrix} \begin{bmatrix} \varepsilon_{x,0} \\ \dots \\ \varepsilon_{x,i} \\ \varepsilon_{y,i} \\ \varepsilon_{z,i} \\ \varepsilon_{s,i} \\ \varepsilon_{r,i} \\ \varepsilon_{p,i} \\ \dots \\ \varepsilon_{p,n} \end{bmatrix} \tag{15}$$

For each link i, i between 1 and n, the following linear combinations are always valid for a manipulator described by the DN notation [6]:

$$\mathbf{J}_{z,(i-1)} \equiv \sin\alpha_i \mathbf{J}_{y,i} + \cos\alpha_i \mathbf{J}_{z,i} \tag{16}$$

$$\mathbf{J}_{r,(i-1)} \equiv a_i \cos\alpha_i \mathbf{J}_{y,i} - a_i \sin\alpha_i \mathbf{J}_{z,i} + \sin\alpha_i \mathbf{J}_{s,i} + \cos\alpha_i \mathbf{J}_{r,i} \tag{17}$$

If joint i is prismatic, then additional combinations of the columns of $\mathbf{J}_e$ are valid:

$$\mathbf{J}_{x,(i-1)} \equiv \mathbf{J}_{x,i} \tag{18}$$

$$\mathbf{J}_{y,(i-1)} \equiv \cos\alpha_i \mathbf{J}_{y,i} - \sin\alpha_i \mathbf{J}_{z,i} \tag{19}$$

The linear combinations shown above are always valid, independently of the values of a_i and α_i. If both position and orientation are measured, then Equations (16)-(19) are the only linear combinations for link i.

To obtain the non-singular Identification Jacobian matrix, called here $\mathbf{G}_e$, columns $\mathbf{J}_{z,(i-1)}$ and $\mathbf{J}_{r,(i-1)}$ must be eliminated from the matrix $\mathbf{J}_e$ for all values of i between 1 and n. If joint i is prismatic, then columns $\mathbf{J}_{x,(i-1)}$ and $\mathbf{J}_{y,(i-1)}$ must also be eliminated. For an n DOF (Degrees Of Freedom) manipulator with r rotary joints and p (p equal to n-r) prismatic joints, a total of $2r$+$4p$ columns are eliminated from the Identification Jacobian $\mathbf{J}_e$ to form its submatrix $\mathbf{G}_e$. This means that $2r$+$4p$ generalized errors cannot be obtained by measuring the end-effector position.

The dependent error parameters eliminated from ε do not affect the end-effector error, resulting in the identity:

$$\mathbf{\Delta X} = \mathbf{J}_e\, \boldsymbol{\varepsilon} \equiv \mathbf{G}_e\, \boldsymbol{\varepsilon}^{*} \tag{20}$$

Using the above identity and the linear combinations of the columns of $\mathbf{J}_e$ from Eqs.(16)-(19), it is possible to obtain all relationships between the generalized error set ε and its independent subset, ε^*. If joint i is revolute (i between 1 and n), then the generalized errors $\varepsilon_{z,(i-1)}$ and $\varepsilon_{r,(i-1)}$ are eliminated, and its values are incorporated into the independent error parameters $\varepsilon^*_{y,i}$, $\varepsilon^*_{z,i}$, $\varepsilon^*_{s,i}$ and $\varepsilon^*_{r,i}$:

$$\begin{cases} \varepsilon^*_{y,i} \equiv \varepsilon_{y,i} + \varepsilon_{z,(i-1)} \sin\alpha_i + \varepsilon_{r,(i-1)} \cdot a_i \cos\alpha_i \\ \varepsilon^*_{z,i} \equiv \varepsilon_{z,i} + \varepsilon_{z,(i-1)} \cos\alpha_i - \varepsilon_{r,(i-1)} \cdot a_i \sin\alpha_i \\ \varepsilon^*_{s,i} \equiv \varepsilon_{s,i} + \varepsilon_{r,(i-1)} \sin\alpha_i \\ \varepsilon^*_{r,i} \equiv \varepsilon_{r,i} + \varepsilon_{r,(i-1)} \cos\alpha_i \end{cases} \tag{21}$$

If joint i is prismatic, then the translational errors $\varepsilon_{x,(i-1)}$ and $\varepsilon_{y,(i-1)}$ are eliminated, and their values are incorporated into the independent error parameters $\varepsilon^*_{x,i}$, $\varepsilon^*_{y,i}$ and $\varepsilon^*_{z,i}$. In this case, Eq.(21) becomes:

$$
\begin{cases}
\varepsilon^{*}_{x,i} \equiv \varepsilon_{x,i} + \varepsilon_{x,(i-1)} \\
\varepsilon^{*}_{y,i} \equiv \varepsilon_{y,i} + \varepsilon_{y,(i-1)} \cos\alpha_i + \varepsilon_{z,(i-1)} \sin\alpha_i + \varepsilon_{r,(i-1)} \cdot a_i \cos\alpha_i \\
\varepsilon^{*}_{z,i} \equiv \varepsilon_{z,i} - \varepsilon_{y,(i-1)} \sin\alpha_i + \varepsilon_{z,(i-1)} \cos\alpha_i - \varepsilon_{r,(i-1)} \cdot a_i \sin\alpha_i \\
\varepsilon^{*}_{s,i} \equiv \varepsilon_{s,i} + \varepsilon_{r,(i-1)} \sin\alpha_i \\
\varepsilon^{*}_{r,i} \equiv \varepsilon_{r,i} + \varepsilon_{r,(i-1)} \cos\alpha_i
\end{cases}
\tag{22}
$$

If the vector ε^{*} containing the independent errors is constant, then the matrix $\mathbf{G}_e$ can be used to replace $\mathbf{J}_e$ in Eq.(20), and Eq.(22) is applied to calculate the estimate of the independent generalized errors ε^{*}, completing the identification process. However, if non-geometric factors are considered, then it is necessary to further model the parameters of ε^{*} as a function of the system configuration prior to the identification process.

2.6. Physical Interpretation of the Redundant Errors

To fully understand the concepts of redundant errors it is a good idea to develop a geometric interpretation.

Equation (16) shows that a translation along the *z(i-1)* axis provokes the same translation error on the end effector as a combination of 2 errors in *z* and *y* direction of the joint z_i. This principle is shown in figure 7.

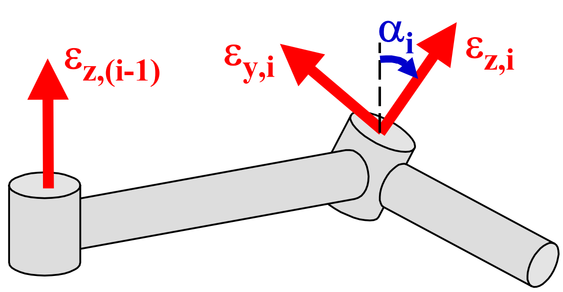

Figure 7 – Combination of translational linear errors [6]

According to Eq.(17) there exists a relation between the rotation error along the z-axis of frame *i-1* and combination of rotational and translational errors along the y and z-axis of frame *i*. Figure 7 gives a simplified interpretation of the rather complex equation.

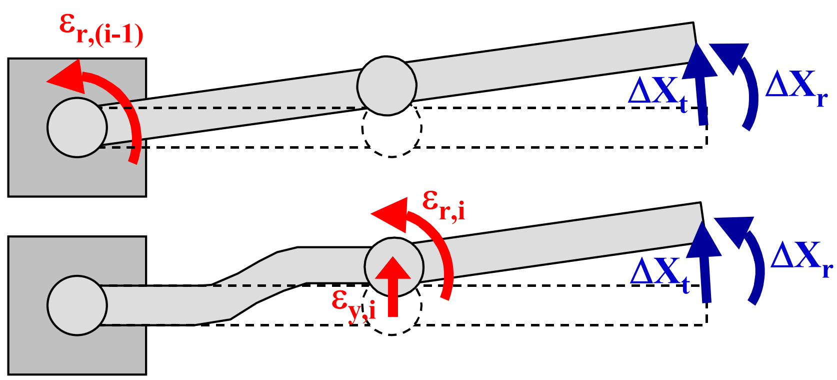

Figure 8 – Simplified combination of error [6]

2.7.
Partial Measurement of End-Effector Pose

When the calibration process is performed using only the end effector position and not the orientation, there are certain columns of the Identification Jacobian matrix $\mathbf{J}_e$ that has to be eliminated since they have no effect on the end-effector position and therefore can't be estimated. The last 3 columns of J_e can be eliminated since they only affect the orientation of the end-effector [6]:

$$\mathbf{J}_{s,n} \equiv \mathbf{J}_{r,n} \equiv \mathbf{J}_{p,n} \equiv \mathbf{0} \tag{23}$$

If the last joint is revolute and its link length a_n is zero, then three additional linear combinations are present:

$$\mathbf{J}_{s,(n-1)} \equiv d_n \, \mathbf{J}_{x,(n-1)} \tag{24}$$

$$\mathbf{J}_{p,(n-1)} \equiv - d_n \, \mathbf{J}_{y,(n-1)} \tag{25}$$

$$\mathbf{J}_{r,(n-1)} \equiv \mathbf{0} \tag{26}$$

This means that the effects of $\varepsilon_{s,(n-1)}$ and $\varepsilon_{p,(n-1)}$ cannot be distinguished from the ones caused by $\varepsilon_{x,(n-1)}$ and $\varepsilon_{y,(n-1)}$, and also the generalized error $\varepsilon_{r,(n-1)}$ is not obtainable. If both link length a_n and joint offset d_n are zero, then the origin of frames n-1 and n coincide at the end-effector position. In this case, Eqs.(23)-(26) can be recursively applied to frames n-1, n-2, and so on, as long as the origin of these frames all lie at the end-effector.

When only a_n is zero, $\varepsilon_{s,(n-1)}$ and $\varepsilon_{y,(n-1)}$ can be eliminated and their values transferred to $\varepsilon_{x,(n-1)}$ and $\varepsilon_{p,(n-1)}$ respectively shown in Eq.(27):

$$\begin{cases} \varepsilon^*_{x,(n-1)} = \varepsilon_{x,(n-1)} + \varepsilon_{s,(n-1)} \cdot d_n \\ \varepsilon^*_{y,(n-1)} = \varepsilon_{y,(n-1)} - \varepsilon_{p,(n-1)} \cdot d_n \end{cases} \quad (27)$$

2.8. Inverse Kinematics

Through the Denavit-Hartenberg notation the direct kinematics of the manipulator is obtained, where the orientation and position of the end-effector is estimated using the angles of all joints as inputs.

Inverse kinematics uses the DN-notation to generate the required angles of the joints to reach the desired position and orientation. In order to be able to automate certain tasks, finding the inverse kinematics is a necessary step.

The inverse kinematics is generally more complex than direct kinematics, and there exist several solutions for the same desired position. For some positions there might not be any solution at all. Due to the non-linearity of the equations it is difficult to elaborate a general solution. Every case has to be estimated individually from the direct kinematic equations. If an analytical solution is not found, it might be found by numerical methods.

In order to be able to reach any desired position and orientation, the manipulator needs at least six degrees of freedom. Manipulators with less degrees of freedom will not be able to reach an arbitrary position in 3D space. If a manipulator has more than six degrees of freedom, there may be an infinite number of possible solutions for every position.

2.8.1. Solvability

In cases where an analytical solution for the inverse kinematics does not exist, iterative numerical solutions there can be implemented. This is generally computationally heavy. An analytical solution is therefore desirable whenever possible.

The existence of an analytical solution depends on the kinematic structure of the manipulator. The structure is usually designed in order to achieve this, so that complex calculations can be avoided.

Reference [10] shows that having three rotational consecutive joints whose axes intersect in one single point for all configurations of the joints is enough to guarantee an analytical solution for a manipulator with six degrees of freedom. This is the case for the manipulator studied in this work. The axes of the joints 4, 5 and 6 intersect at the origin of joints 4 and 5.

2.9. Experimental Procedures

In the calibration procedure the desired reference frame is the robot base frame. The actual robot base is defined to be the coordinate on the rotation axis of joint 1 with the closest relative distance to the rotation axis of joint 2. The end-effector position is measured using for example a laser tracker. The robot base frame needs to be estimated relative to the reference frame of the laser tracker. This assures that the absolute accuracy of the robot can be measured relative to its base axes. The rotation axes of joints 1 and 2 are estimated by measuring the trajectory of a probe attached to the moveable structure of the manipulator. The joints 1 and 2 are then moved individually and the trajectories of the probe are measured. The methodology used to identify the position and orientation of the robot base is inspired by [11]. The procedure is shown schematically in Figure **9** and Figure **10** for a typical manipulator.

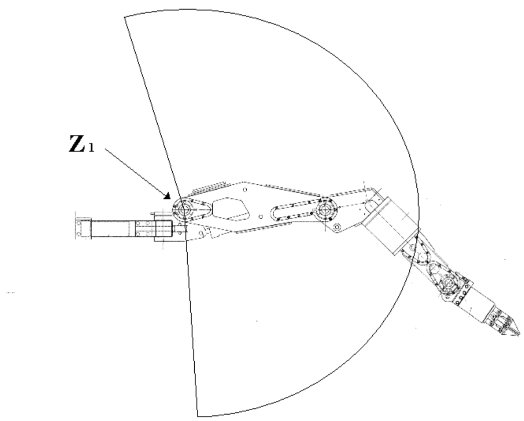

Figure 9 – Finding the rotation axis of joint 2 (Z_1), side view

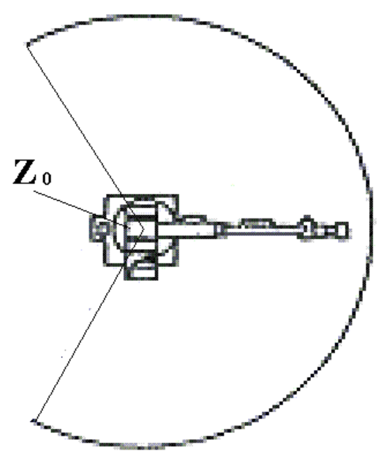

Figure 10 – Finding the rotation axis of joint 1 (Z_0), upper view

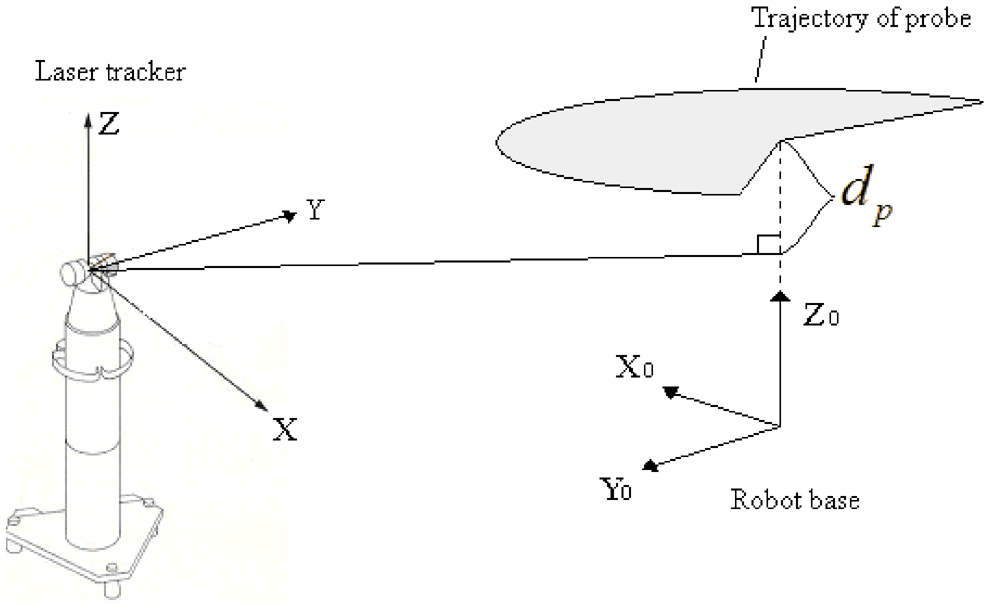

Figure 11 - The trajectory of the probe forms a plane that is found by a least square approximation

The trajectories of the end-effector, when joints 1 and 2 are moved individually, form two arcs that define the rotation axes of these joints. This is shown in Figure 11. To find a mathematic model of the two arcs relative to the laser tracker, the respective planes of the arcs are found using a least square estimate. A plane in space can be defined by the following formula: $a_p x + b_p y + c_p z + d_p = 0$.

Then, defining the matrix C_a, with measured 3D coordinates of an arc (relative to the laser tracker):

$$C_a = \begin{bmatrix} x_{a,1} & y_{a,1} & z_{a,1} & 1 \\ x_{a,2} & x_{a,2} & x_{a,2} & 1 \\ \vdots & \vdots & \vdots & \vdots \\ x_{a,N_a} & y_{a,N_a} & z_{a,N_a} & 1 \end{bmatrix} \tag{28}$$

The least square estimate of the plane is found by :

$$v_p = \begin{bmatrix} v_{p,1} & v_{p,2} & v_{p,3} & v_{p,4} \end{bmatrix}^T = eig\left(C_a^T C_a\right) \tag{29}$$

where v_p is the eigenvector corresponding to the smallest eigenvalue of the 4 x 4 matrix $C_a^T C_a$. The normal plane, shown in Figure 12 is given by:

$$n_p = \begin{bmatrix} a_p & b_p & c_p \end{bmatrix} = \frac{\begin{bmatrix} v_{p,1} & v_{p,2} & v_{p,3} \end{bmatrix}}{\left\| v_{p,1} \quad v_{p,2} \quad v_{p,3} \right\|} \tag{30}$$

The Euclidian distance from the center of the laser tracker reference frame to the plane is given by:

$$d_p = \frac{v_{p,4}}{\left\| v_{p,1} \quad v_{p,2} \quad v_{p,3} \right\|} \tag{31}$$

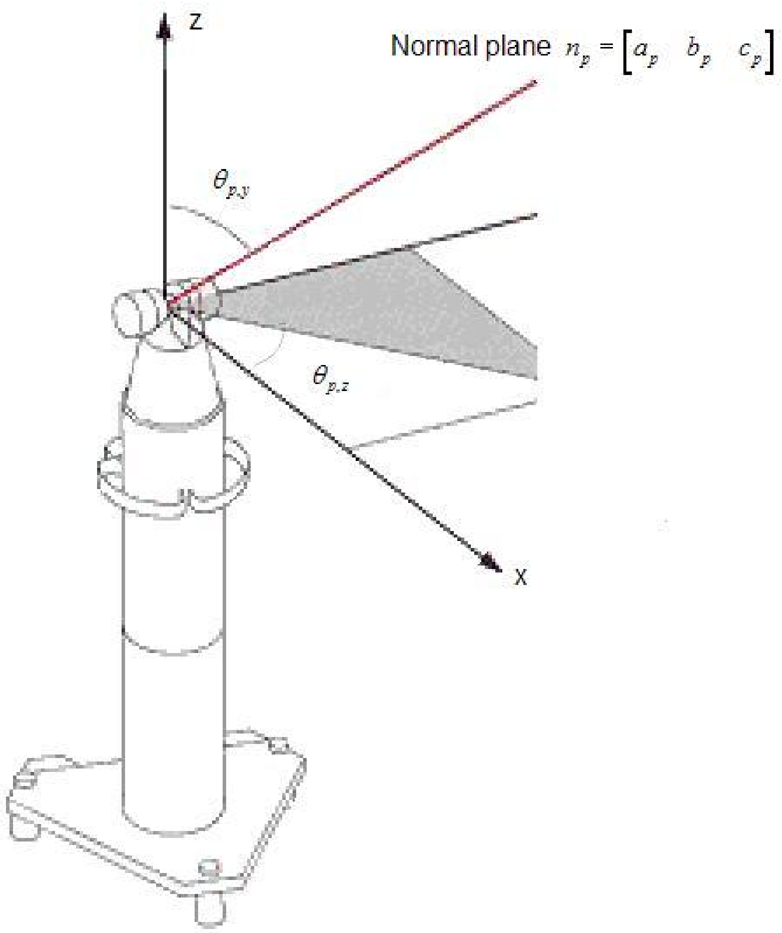

Figure 12 - Angles between the laser tracker reference frame and the normal plane [12]

The next step is to project the arcs' coordinates onto the estimated plane. In order to do this, the angle between the z-axis of the laser trackers' reference frame and the normal plane is estimated. The rotation angle around the y-axis is given by:

$$\theta_{p,y} = \cos^{-1}\left(c_p\right) \tag{32}$$

where $\cos^{-1}$ is the arc cosine function. The rotation around the z-axis is given by.

$$\theta_{p,z} = \tan^{-1}\left(\frac{b_p}{a_p}\right) \tag{33}$$

The rotation matrix between the normal plane and the laser trackers' z-axis is then given by:

$$R_p = \begin{bmatrix} \cos\left(\theta_{p,z}\right) & -\sin\left(\theta_{p,z}\right) & 0 \\ \sin\left(\theta_{p,z}\right) & \cos\left(\theta_{p,z}\right) & 0 \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cos\left(\theta_{p,y}\right) & 0 & \sin\left(\theta_{p,y}\right) \\ 0 & 1 & 0 \\ -\sin\left(\theta_{p,y}\right) & 0 & \cos\left(\theta_{p,y}\right) \end{bmatrix} \tag{34}$$

To find the rotation center of the arc, the coordinate matrix C_a is projected onto the estimated plane, creating the projected coordinate set $C_a^{`}$.

$$C_a^{`} = \left(R_p^T C_a^T\right)^T \tag{35}$$

The new set of coordinates is given relative to the laser tracker, but the reference z-axis is transformed so that it is parallel to the estimated normal plane. This gives the same z-value for all the coordinates. The rotation center of the arc can then be estimated using the x and y coordinates. To estimate the rotation center in the plane, a circle is fitted using the projected coordinates. The circle is fitted using a least square estimate proposed in [13]. This method is much simpler to implement than the iterative Levenberg-Marquardt method used in [11].

The *x* and *y* coordinates of $C_a^{`}$ are extracted. Defining the x coordinates as $x_a^{`}$ and the *y* coordinates as $y_a^{`}$.

Having a set of 2D coordinates, $x_a^{`}$ and $y_a^{`}$ represent the trajectory of the probe projected onto the estimated plane. The mean of the coordinates, respectively, is given by:

$$\overline{x_a} = \frac{1}{N_p}\sum_{i=1}^{N_p} x_{a,i}^{`}, \qquad \overline{y}_p = \frac{1}{N_p}\sum_{i=1}^{N_p} x_{a,i}^{`} \tag{36}$$

where N_p is the number of measured samples.

Introducing the translated variables:

$$x_t = x_a^{`} - \overline{x_a^{`}} \tag{37}$$

and

$$y_t = y_a^{`} - \overline{y_a^{`}} \tag{38}$$

The observation matrix is given by:

$$C_t = \begin{bmatrix} 2x_{t,1} & 2x_{t,1} & -1 \\ : & : & : \\ 2x_{t,N_p} & 2y_{t,N_p} & -1 \end{bmatrix} \tag{39}$$

The vector b_o is given by:

$$b_t = \begin{bmatrix} {x_{t1}}^2 + {y_{t1}}^2 \\ : \\ {x_{tN_p}}^2 + {y_{tN_p}}^2 \end{bmatrix} \tag{40}$$

Let the vector u be the least square solution to the problem $A \cdot u = b$ giving:

$$u_t = \left(C_t^T C_t\right)^{-1} \cdot C_t^T \cdot b_t \tag{41}$$

So u_t is then a 3 x 1 vector: $u_t = \begin{bmatrix} u_1 & u_2 & u_3 \end{bmatrix}^T$.

Defining x_r and y_r as the coordinates of the circle center, and r_r as the radius of the circle. They are given by the following equations:

$$x_r = u_1 \tag{42}$$

$$y_r = u_2 \tag{43}$$

$$r_r = \sqrt{{x_r}^2 + {y_r}^2 - u_3} \tag{44}$$

The rotation center of the joint relative to the laser tracker is then given by:

$$c_r = R_p \cdot \begin{bmatrix} x_r & y_r & d_p \end{bmatrix}^T \tag{45}$$

Estimating the rotation center and normal plane for both the joints 1 and 2, the actual robot base can be found using the triangulation method shown in section 3.5.