

J. M. S. T. Motta
 Universidade de Brasília
 Faculdade de Tecnologia
 Departamento de Engenharia Mecânica
 70910-900 Brasília, DF, Brazil
 jmmotta@unb.br

An Investigation of Singularities in Robot Kinematic Chains Aiming at Building Robot Calibration Models for Off-line Programming

Robot Calibration is a term applied to the procedures used in determining actual values that describe the geometric dimensions and mechanical characteristics of a robot or multibody structure. A robot calibration system must consist of appropriate robot modeling techniques, accurate measurement equipment, and reliable model parameter determination methods. For practical improvement of a robot's absolute accuracy, error compensation methods are required that use calibration results. Important to robot calibration methods is an accurate kinematic model that has identifiable parameters. This parameterized kinematic model must be complete, continuous and minimal. This work concerns to the implementation of techniques to optimize kinematic models for robot calibration through numerical optimization of the mathematical model. The optimized model is then used to compensate the model errors in an off-line programming system, enhancing significantly the robot kinematic model accuracy. The optimized model can be constructed in an easy and straight operation, through automatic assignment of joint coordinate systems and geometric parameter to the robot links. Assignment of coordinate systems by this technique avoids model singularities that usually spoil robot calibration results.

Keywords: *Off-line programming, model optimization, robot calibration, kinematic models*

Introduction

Off-line programming is, by definition, the technique of generating a robot program without using a real machine. It presents several advantages over the on-line method. However, there are inevitably differences between the computer model used to perform the graphic simulation and the real world. This is because the effective use of off-line programming in industrial robots requires, additionally, a knowledge of tolerances of the manufacturing components in order to enable realistic planning, i.e. to reduce the gap between simulation and reality. In an actual robot system programming this is still a relatively difficult task and the generated programs still require manual on-line modification at the shop floor. A typical welding line with 30 robots and 40 welding spots per robot takes about 400 hours for robot teaching (Bernhardt, 1997). The difficulties are not only in the determination of how the robot can perform correctly its function, but also for it to be able to achieve accurately a desired location in the workspace. Robot pose errors are attributed to several sources, including the constant (or configuration-independent) errors in parameters (link lengths and joint offsets), deviations which vary predictably with position (e.g., compliance, gear transmission errors) and random errors (e.g., due to the finite resolution of joint encoders). Constant errors are referred to as geometric errors and variable errors are referred to as non-geometric errors (Roth, Mooring and Ravani, 1987). According to Bernhardt (1997) and Schröer (1993), constant errors represent approximately 90% of the overall robot pose errors. Industrial robots usually show pose errors from about 5 to 15mm, even when they are new, and after proper calibration these error can be reduced to about less than 0.5mm (Bernhardt, 1997, Motta, Carvalho and McMaster, 2001).¹

The Robot Calibration problem has been investigated for more than one decade, but many of its obstacles are still around. Usually, one can tackle the problem implementing model or modeless methods. Modeless methods does not need any kinematic model,

using only a grid of known points located in the robot workspace as a standard calibration board. The robot is moved through all the grid points, and the position errors are stored for future compensation by using a bilinear interpolation method and polynomial fitting (Zhuang & Roth, 1996, Park, Xu and Mills, 2002) or error mapping (Bai and Wang, 2004). Although modeless methods are simpler, the calibrated workspace region is small, and each time the robot is to work off that region the calibration process has to be repeated. On the other side, model methods allow a large workspace region to be calibrated, leading to full model calibration. However, important to model methods is an accurate kinematic model that is complete, minimal and continuous and has identifiable parameters (Schröer, Albright and Grethlein, 1997). Researchers have used specific kinematic models that depend on a particular robot geometry and/or calibration method. Model identifiability has already been addressed (e.g., Everett and Hsu, 1988, Zhuang, 1992), and Motta and McMaster (1999) and Motta, Carvalho e McMaster (2001) have shown experimental and simulation results using a rational technique to find an optimal model for a specific joint configuration, requiring a few number of measurement points (for a 6 DOF robot only 15 measurement points) for a model with only geometric parameters (30), in opposition to hundreds of measurement points claimed by other authors (Drouet et al., 2002, Park, Xu and Mills, 2002). A model with singularities or quasi-singular parameterization turns the identification process to be ill-conditioned, leading to solutions that cannot satisfy accuracy requirements when the manipulator is to move off the measurement points. Furthermore, time to convergence increases or may not exist convergence at all, and the number of measurement points may be ten-fold larger than the necessary (Motta, 1999).

The objective of this article is to investigate robotic model singularities, discuss a method to construct kinematic models for robot calibration assuring model continuity, completeness and minimality and to present a computer program to carry out this task automatically before the identification step. The computer program was developed using the API (Application Programming Interface) JAVA3D computer language. This language encompasses three important needs of off-line programming software: support to computer graphics, serial communication and TCP/IP

¹Paper accepted March, 2005. Technical Editor: Edgar Nobuo Mamiya.

communication. The computer program allows one to build an optimized model for any kinematic chain, providing a powerful tool for a reliable and a speeded-up robot calibration process. Further steps to a full robot calibration process, i.e. measurement, identification and compensation will not be discussed in this article. The reader can find further discussions in Motta and Sousa (2004), Motta and McMaster (1999), Schröer (1993) and Roth, Mooring and Ravani (1987).

A Singularity-Free Approach for Kinematic Models

Single minimal modeling convention that can be applied uniformly to all possible robot geometries cannot exist owing to fundamental topological reasons concerning mappings from Euclidean vectors to spheres (Schröer, 1993). However, after investigating many topological problems in robots, concerning inverse kinematics and singularities, Baker (1990) suggested that the availability of an assortment of methods for determining whether or not inverse kinematic functions can be defined on various subsets of the operational spaces would be useful, but even more important, a collection of methods by which inverse functions can actually be constructed in specific situations. Another insightful paper about robot topologies was published by Gottlieb (1986), who noted that inverse functions can never be entirely successful in circumventing the problems of singularities when pointing or orienting.

Mathematically, model-continuity is equivalent to continuity of the inverse function T^{-1} , where T is the product of elementary transformations (rotation and translation) between joints. From this, the definition of parameterization's singularity can be stated as a transformation $T_s \in E$ (parameterization's space of the Euclidean Group - 3 rotations and 3 translations), where the parameter vector $p \in R^6$ (p represents distance or angle) exists such that the rank of the Jacobian $J_s = dT_s/dp$ is smaller than 6. In other way, each parameterization T can be investigated concerning their singularities detecting the zeroes of determinant $\det(J^T J)$ considered as a function of parameter p .

Thus, investigating the main kinematic modeling conventions one can represent the transformation between links in the Euclidean Group as

$$T = T_x(p_x).T_y(p_y).T_z(p_z).R_z(\gamma).R_y(\beta).R_x(\alpha) \tag{1}$$

where p_x, p_y, p_z are translation coordinates and α, β, γ are rotation coordinates for the axes x, y and z respectively. Then,

$$T = \begin{bmatrix} C\gamma.C\beta & C\gamma.S\beta.S\alpha - S\gamma.C\alpha & C\gamma.S\beta.C\alpha + S\gamma.S\alpha & P_x \\ S\gamma.C\beta & S\gamma.S\beta.S\alpha + C\gamma.C\alpha & S\gamma.S\beta.C\alpha - C\gamma.S\alpha & P_y \\ -S\beta & C\beta.S\alpha & C\beta.C\alpha & P_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2}$$

where $C=\cos(\)$ and $S=\sin(\)$.

In a more simple symbolic form it can be represented as

$$T = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{3}$$

which after the decomposition of the singularities through the Jacobian determinant results in a manifold of singularities

$$T_s = \begin{bmatrix} 0 & o_x & a_x & p_x \\ 0 & o_y & a_y & p_y \\ \pm 1 & 0 & 0 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{4}$$

This manifold represents $\gamma = \pm \pi/2$ and $\beta = \pm \pi/2$ (eq. 2), which means there is a singularity when y and z axes are parallel.

The Denavit-Hartenberg convention (parameterization) (Paul, 1981) leads to an elementary transformation represented by

$$T(\theta, p_z, p_x, \alpha) = R_z(\theta).T_z(p_z).T_x(p_x).R_x(\alpha) \tag{5}$$

Following the same procedure as before the manifold of singularities is

$$T_s = \begin{bmatrix} n_x & o_x & 0 & p_x \\ n_y & o_y & 0 & p_y \\ n_z & o_z & \pm 1 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{6}$$

This result consists of all elements represented as parallel rotary joints. This can be verified by observing the third column showing the representation of one joint axis (z) into the previous one.

The Hayati-convention (Hayati & Mirmirani, 1985) can be represented by

$$T(\theta, p_x, \alpha, \beta) = R_z(\theta).T_x(p_x).R_x(\alpha).R_y(\beta) \tag{7}$$

There are two manifolds of singularities,

$$T_s = \begin{bmatrix} n_x & o_x & a_x & p_x \\ n_y & o_y & a_y & p_y \\ n_z & o_z & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \text{ OR } T_s = \begin{bmatrix} n_x & o_x & \lambda p_x & p_x \\ n_y & o_y & \lambda p_y & p_y \\ n_z & o_z & \lambda p_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{8}$$

These representations show that if the distal z axis is in the same x - y plane of the proximal coordinate system (so perpendicular) or points to its origin, then there is a singularity.

The Veitschegger convention (Veitschegger & Wu, 1986) is a 5-dimensional parameterization as

$$T = R_z(\theta).T_z(p_z).T_x(p_x).R_x(\alpha).R_y(\beta) \tag{9}$$

The manifolds of singularities are

$$T_s = \begin{bmatrix} n_x & o_x & a_x & 0 \\ n_y & o_y & a_y & 0 \\ n_z & o_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \text{ OR } T_s = \begin{bmatrix} n_x & o_x & a_x & \lambda a_x \\ n_y & o_y & a_y & \lambda a_y \\ n_z & o_z & 0 & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{10}$$

These representations show that if the origin of the distal joint coordinate system is on the z -axis of the proximal one, or the z axes are perpendicular and intercept each other, then there is a singularity. However this convention is not minimal.

Using the same technique presented so far for prismatic joints sets of parameterizations can be used so fourth. The results can be outlined in a list together with their application ranges. The set is a complete, minimal and model-continuous kinematic model (Schröer, Albricht and Grethlein, 1997). The transformations are

modeled from a current frame C to a revolute joint, J_R , or to a prismatic joint, J_P , or a fixed and arbitrarily located target frame, TCP-frame. Some of them are not unique since other solutions can fulfill the requirements. Additional elementary transformations are possible, but any added elementary transformation is redundant and thus, cannot be identified. A non-expanded model can always be found, which describes the same kinematic structure as the expanded model.

Elementary transformations that are part of the complete, minimal and model-continuous sub-model being identified are marked bold, i.e. those elementary transformations including model parameters to be identified (errors): (symbols \perp and \parallel mean orthogonal and parallel axes respectively)

- Transformation from robot base frame (B) to first joint where joint is translational (J_T):

$$B \perp J_T: P = (T_X, T_Y, T_Z, \mathbf{R}_Z, \mathbf{R}_X) \quad (11)$$

$$B \parallel J_T: P = (T_X, T_Y, T_Z, \mathbf{R}_X, \mathbf{R}_Y) \quad (12)$$

And, if first joint is rotational (J_R):

$$B \perp J_R: P_X = (T_Y, T_Z, \mathbf{R}_Z, \mathbf{R}_X) \quad (13)$$

(If joint axis is near x -axis of frame B)

$$P_Y = (T_X, T_Z, \mathbf{R}_Z, \mathbf{R}_X) \quad (14)$$

(If joint axis is near y -axis of frame B)

$$B \parallel J_R: P_Z = (T_X, T_Y, \mathbf{R}_X, \mathbf{R}_Y) \quad (15)$$

- Transformations between consecutive joint frames:

$$J_R \perp J_R: P = (\mathbf{R}_Z, T_Z, T_X, \mathbf{R}_X, T_Z) \quad (16)$$

(Denavit-Hartenberg parametrization)

$$J_R \parallel J_R: P = (\mathbf{R}_Z, T_X, \mathbf{R}_X, \mathbf{R}_Y, T_Z) \quad (17)$$

(assumption: joint axes are not identical)
(Hayati parametrization)

$$J_T \perp J_R: P = (T_Z, \mathbf{R}_Z, T_X, \mathbf{R}_X, T_Z) \quad (18)$$

$$J_T \parallel J_R: P = (T_Z, T_X, T_Y, \mathbf{R}_X, \mathbf{R}_Y, T_Z) \quad (19)$$

$$J_T \perp J_T: P = (T_Z, T_X, T_Y, \mathbf{R}_Z, \mathbf{R}_X) \quad (20)$$

$$J_T \parallel J_T: P = (T_Z, T_X, T_Y, \mathbf{R}_X, \mathbf{R}_Y) \quad (21)$$

$$J_R \perp J_T: P = (\mathbf{R}_Z, T_X, T_Y, T_Z, \mathbf{R}_X) \quad (22)$$

$$J_R \parallel J_T: P = (R_Z, T_X, T_Y, T_Z, \mathbf{R}_X, \mathbf{R}_Y) \quad (23)$$

Transformation from last joint to TCP (Tool Center Point) :

$$J_T \perp TCP: P = (T_Z, T_Y, T_X, [\mathbf{R}_Z, \mathbf{R}_Y, \mathbf{R}_Z]) \quad (24)$$

$$J_T \parallel TCP: P = (T_Z, T_Y, T_X, [\mathbf{R}_Z, \mathbf{R}_Y, \mathbf{R}_X]) \quad (25)$$

$$J_R \perp TCP: P = (R_Z, T_X, T_Y, T_Z, [\mathbf{R}_Z, \mathbf{R}_Y, \mathbf{R}_Z]) \quad (26)$$

$$J_R \parallel TCP: P = (R_Z, T_X, T_Y, T_Z, [\mathbf{R}_Z, \mathbf{R}_Y, \mathbf{R}_X]) \quad (27)$$

Parameters in brackets are not identifiable without TCP-orientation measurements. As an example of the application of the equations above a case using eqs. (16) and (17) is shown below, namely, the known Denavit-Hartenberg (D-H) and Hayati

parameterizations. The equations are referred to Fig. 1. For the D-H case one can define four cases:

1 - $Z_R(2)$ is parallel in the same direction as X_r :

$$P = R_Z(90^\circ).T_Z(p_z).T_X(p_y).R_X(90^\circ).T_Z(p_x) \quad (28)$$

2 - $Z_R(3)$ is parallel in the opposite direction of X_r :

$$P = R_Z(90^\circ).T_Z(p_z).T_X(p_y).R_X(-90^\circ).T_Z(-p_x) \quad (29)$$

3 - $Z_R(4)$ is parallel in the same direction as Y_r :

$$P = R_Z(0^\circ).T_Z(p_z).T_X(p_y).R_X(-90^\circ).T_Z(-p_y) \quad (30)$$

4 - $Z_R(5)$ is parallel in the opposite direction of Y_r :

$$P = R_Z(0^\circ).T_Z(p_z).T_X(p_x).R_X(90^\circ).T_Z(-p_y) \quad (31)$$

For the case of the Hayati parameterization (eq. 17) (valid only for $T_X \neq 0$) if a joint subsequent position needs two translation parameters to be located, (i. e. in the X and Y direction) one of the two has to be vanished, locating the joint frame in a position such that only one remains. Four cases may be assigned:

Supposing only p_x :

1 - $Z_R(0)$ is in the same direction as Z_r :

$$P = R_Z(0^\circ).T_X(p_x).R_X(0^\circ).R_Y(0^\circ).T_Z(p_z) \quad (32)$$

2 - $Z_R(1)$ is in the opposite direction of Z_r :

$$P = R_Z(0^\circ).T_X(p_x).R_X(180^\circ).R_Y(0^\circ).T_Z(-p_z) \quad (33)$$

Supposing only p_y :

3 - $Z_R(0)$ is in the same direction as Z_r :

$$P = R_Z(90^\circ).T_X(p_y).R_X(0^\circ).R_Y(0^\circ).T_Z(p_z) \quad (34)$$

4 - $Z_R(0)$ is in the opposite direction of Z_r :

$$P = R_Z(90^\circ).T_X(p_y).R_X(0^\circ).R_Y(0^\circ).T_Z(-p_z) \quad (35)$$

Figure 1: Frames showing the D-H and Hayati parametrizations

Kinematic Modelling - Assignment of Coordinate Frames

The first step to kinematic modeling is the proper assignment of coordinate frames to each link. Each coordinate system here is orthogonal, and the axes obey the right-hand rule.

For the assignment of coordinate frames to each link one may move the manipulator to its zero position. The zero position of the manipulator is the position where all joint variables are zero. This procedure may be useful to check if the zero positions of the model constructed are the same as those used by the controller, avoiding the need of introducing constant deviations to the joint variables (joint positions).

The first step to assign coordinate frames to joints is to make the z-axis coincident with the joint axis. This convention is used by many authors and in many robot controllers (McKerrow, 1995, Paul, 1981). For a prismatic joint, the direction of the z-axis is in the direction of motion, and its sense is away from the joint. For a revolute joint, the sense of the z-axis is towards the positive direction of rotation around the z-axis. The positive direction of

rotation of each joint can be easily found by moving the robot and reading the joint positions on the robot controller display.

According to McKerrow (1995) and Paul (1981), the base coordinate frame (robot reference) may be assigned with axes parallel to the world coordinate frame. The origin of the base frame is coincident with the origin of joint 1 (first joint). This assumes that the axis of the first joint is normal to the x-y plane. This location for the base frame coincides with many manufacturers' defined base frame.

Afterwards coordinate frames are attached to the link at its distal joint (joint farthest from the base). A frame is internal to the link it is attached to (there is no movements relative to it), and the succeeding link moves relative to it. Thus, coordinate frame i is at joint $i+1$, that is, the joint that connects link i to link $i+1$.

The origin of the frame is placed as following: if the joint axes of a link intersect, then the origin of the frame attached to the link is placed at the joint axes intersection; if the joint axes are parallel or do not intersect, then the frame origin is placed at the distal joint; subsequently, if a frame origin is described relative to another coordinate frame by using more than one direction, then it must be moved to make use of only one direction if possible. Thus, the frame origins will be described using the minimum number of link parameters.

The x-axis or the y-axis have their direction according to the convention used to parameterize the transformations between links (e.g. equations 16 to 23). At this point the homogeneous transformations between joints must have already been determined. The other axis (x or y) can be determined using the right-hand rule.

A coordinate frame can be attached to the end of the final link, within the end-effector or tool, or it may be necessary to locate this coordinate frame at the tool plate and have a separate hand transformation. The z-axis of the frame is in the same direction as the z-axis of the frame assigned to the last joint ($n-1$).

The end-effector or tool frame location and orientation is defined according to the controller conventions. Geometric parameters of length are defined to have an index of joint and direction. The length p_{ni} is the distance between coordinate frames $i-1$ and i , and n is the parallel axis in the coordinate system $i-1$. Figures 2 and 3 shows the above rules applied to a PUMA-560 and a ABB IRB-2400 robots with all the coordinate frames and geometric features, respectively.

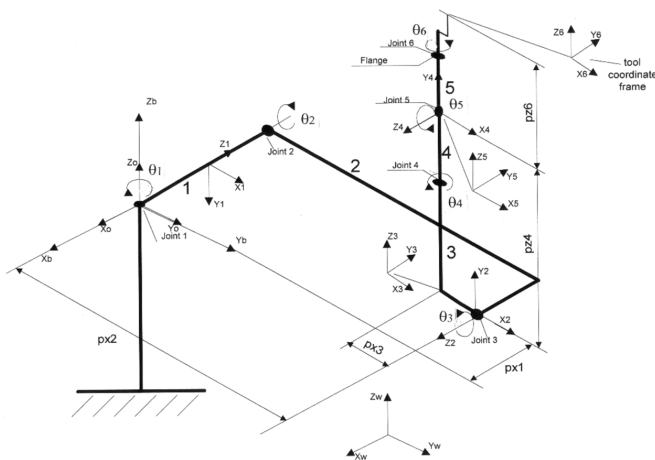


Figure 2. Skeleton of the PUMA 560 Robot with coordinate frames in the zero position and geometric variables for kinematic modeling. (Out of scale).

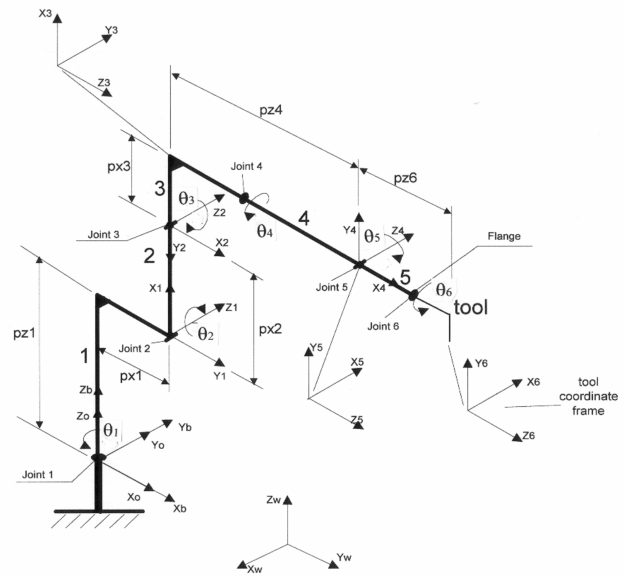


Figure 3. Skeleton of the ABB IRB-2400 Robot with coordinate frames in the zero position and geometric variables for kinematic modeling. (Out of scale).

Results

The computer program for construction of kinematic models for robot calibration was developed in API-JAVA3D language. Figure 4 illustrates the input window for entering model parameter data. As shown in the figure, the system needs 3 types of information about the robot joints. The first is concerned to the type of joint: prismatic or revolute. The second refers to the position of the joint coordinate system origin relative to the previous joint system, measured according to the Base Coordinate System (BCS). The last input information is the joint axis orientation (z axis), measured relative to the BCS.

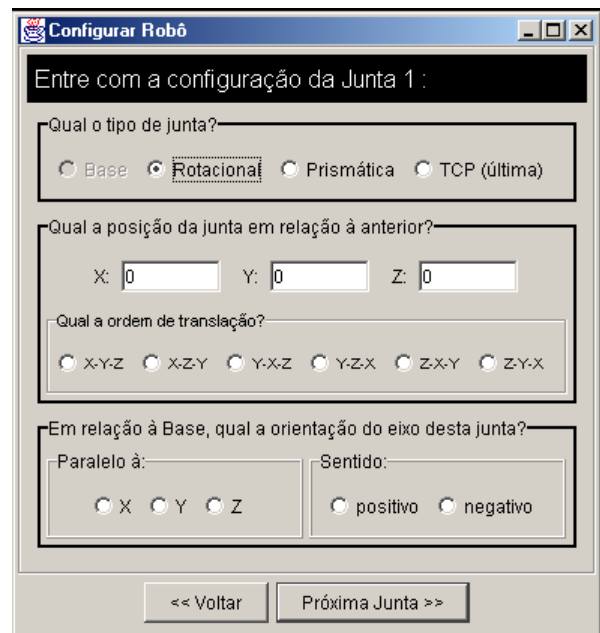


Figure 4. Input data window for entering robot model joint positions.

Model joint data are entered one by one up to the TCP (Tool Center Position), which is the last point in the model geometric sequence.

Figure 5 and 6 show the final models obtained by the computer program, showing the complete robot model for the PUMA-560 and the ABB IRB-2400 robots on the screen. All kinematic parameters and homogeneous transformations are already determined at this stage. The model is then ready for the parameter identification step.

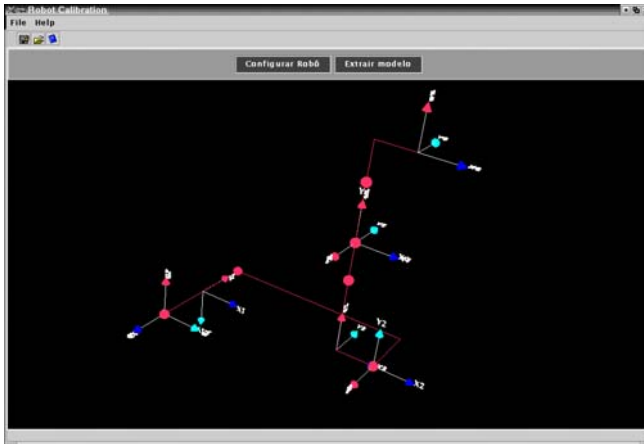


Figure 5. PUMA 560 Robot model obtained by the software.

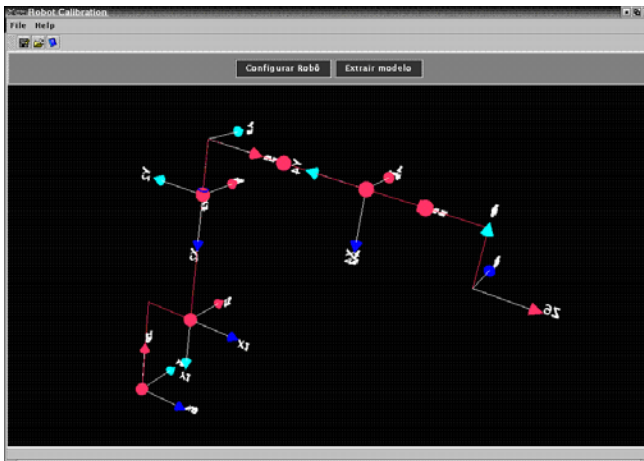


Figure 6. ABB IRB-2400 Robot model obtained by the software.

Comparing the models shown in Fig. 2 and Fig. 3 to the models shown in Fig. 5 and Fig. 6 respectively it is easy to see that they are the same.

The software was tested with many different robot systems as IRB-2000, IRB-6400 and specially designed ones. Up to date all tests were well succeeded.

Conclusions

It was discussed a method to construct optimized kinematic models specifically for robot calibration, i. e. a model where geometric error parameters may be assigned to avoiding singular configurations. The model can be easily constructed joint by joint in a progressive way. The rules adopted to select proper joint coordinate systems in the kinematic model are based on a singularity-free approach that avoids ill-conditioned mathematical

models in the identification process. The system has been tested in many robot topologies and results show coincidence between the computer program outputs and predicted models. So the way is paved for a complete robot calibration system encompassing modeling, data acquisition from measurements, parameter identification and model compensation, bringing about a system that can be linked to robot off-line programming systems, allowing considerable improvements to the accuracy of this type of robot programming.

References

- Bai, Y and Wang, D., 2004, "Improve the Robot Calibration Accuracy Using a Dynamic Online Fuzzy Error Mapping System", *IEEE Transactions on System, Man, And Cybernetics – Part B: Cybernetics*, Vol. 34, No. 2, pp. 1155-1160.
- Baker, D.R., 1990, "Some topological problems in robotics", *The Mathematical Intelligencer*, Vol.12, No.1, pp. 66-76.
- Bernhardt, R., 1997, "Approaches for commissioning time reduction", *Industrial Robot*, Vol. 24, No. 1, pp. 62-71.
- Drouet, Ph., Dubowsky, S., Zeghloul, S. and Mavroidis, C., 2002, "Compensation of geometric and elastic errors in large manipulators with an application to a high accuracy medical system", *Robotica*, Vol. 20, pp. 341-352.
- Everett, L.J. and Hsu, T-W, 1988, "The Theory of Kinematic Parameter Identification for Industrial Robots", *Transaction of ASME*, No. 110, pp. 96-100.
- Gottlieb, D.H., 1986, "Robots and Topology", *Proceedings of the IEEE International Conference on Robotics and Automation*, pp. 1689-1691.
- Hayati, S. and Mirmirani, M., (1985), "Improving the Absolute Positioning Accuracy of Robots Manipulators", *Journal of Robotic Systems*, Vol. 2, No. 4, pp. 397-413.
- McKerrow, P. J. (1995), "Introduction to Robotics", 1st ed., Ed. Addison Wesley, Singapore.
- Motta, J. M. S. T., 1999, "Optimised Robot Calibration Using a Vision-Based Measurement System with a Single Camera", Ph.D. thesis, School of Industrial and Manufacturing Science, Cranfield University, UK.
- Motta, J. M. S. T. and McMaster, R. S., 1999, "Modeling, Optimizing and Simulating Robot Calibration with Accuracy Improvements", *Journal of the Brazilian Society of Mechanical Sciences*, Vol. 21, No. 3, pp. 386-402.
- Motta, J. M. S. T., Carvalho, G. C. and McMaster, R. S., 2001, "Robot Calibration Using a 3-D Vision-Based Measurement System With a Single Camera", *Robotics and Computer Integrated-Manufacturing*, Ed. Elsevier Science, U.K., v. 17, n. 6, p. 457-467.
- Motta, J. M. S. T., Sousa, C. A. G., 2004, "A Dedicated Welding Off-line Programming System with Robot Calibration Accuracy Improvements", *Revista Soldagem e Inspeção*, São Paulo, Brazil, V. 9, n. 2, pp. 96-102.
- Park, E. J., Xu, W and Mills, J. K., 2002, "Calibration-based absolute localization of parts for multi-robot assembly", *Robotica*, Vol. 20, pp. 359-366.
- Paul, R. P., (1981), "Robot Manipulators - Mathematics, Programming, and Control", Boston, MIT Press, Massachusetts, USA.
- Roth, Z.S., Mooring, B.W. and Ravani, B., 1987, "An Overview of Robot Calibration", *IEEE Journal of Robotics and Automation*, RA-3, No. 3, pp. 377-85.
- Schröer, K., 1993, "Theory of kinematic modeling and numerical procedures for robot calibration", *Robot Calibration*, edited by R. Bernhardt and S. L. Albright, Chapman & Hall, London.
- Schröer, K., Albright, S. L. and Grethlein, M., 1997, "Complete, Minimal and Model-Continuous Kinematic Models for Robot Calibration", *Robotics & Computer-Integrated Manufacturing*, Vol. 13, No. 1, pp. 73-85.
- Veitschegg, K. W. and Wu, C. W., (1986), "Robot Accuracy Analysis based on Kinematics". *IEEE Journal of Robotics and Automation*, Vol. 2, No. 3, pp. 171-179.
- Zhuang, H., 1992, "A Complete and Parametrically Continuous Kinematic Model for Robot Manipulators", *IEEE Transactions on Robotics and Automation*, Vol. 8, no. 4, pp. 451-63.
- Zhuang, H. and Roth, Z. S. (1996), "Camera-Aided Robot Calibration", 1st ed., CRC Press, Boca Raton, Fla, USA.