Theoretical and Practical Aspects of Robot Calibration with Experimental Verification

Luciano Selva Ginani
luciano.ginani@tu-ilmenau.de
TU – Ilmenau
Institut für Maschinen- und Gerätekonstruktion
Ilmenau, Germany

José Maurício S. T. Motta
jmmotta@.unb.br
University of Brasilia
Department of Mechanical Engineering
70910-900 Brasília, DF, Brazil

One of the greatest challenges in today's industrial robotics is the development of off-line programming systems that allow drastic reduction in robots' reprogramming time, improving productivity. The article purpose is to pave the way to the construction of generic calibration systems easily adapted to any type of robot, regardless their application, such as modular robots and robot controllers specifically designed for non-standard applications. A computer system was built for developing and implementing a calibration system that involves the joint work of computer and measurement systems. Each step of this system's development is presented together with its theoretical basis. With the development of a remote maneuvering system based on ABB S3 controller experimental tests have been carried out using an IRB2000 robot and a measurement arm (ITG ROMER) with 0.087 mm of position measurement accuracy. The robot model used by its controller was identified and the robot was calibrated and evaluated in different workspaces resulting in an average accuracy improvement from 1.5 mm to 0.3 mm.
Keywords: *robot calibration, off-line programming, parameter identification, robot kinematic models*

Introduction

For decades robots have been used in manufacturing industries to replace men work in simple, repetitive and dangerous tasks. Investments on the robotic research and technological achievements in computer sciences and electronics have led to new possibilities to make robots more accurate and precise, pushing the field of robotics towards an enormous amount of applications, from industry to service, entertainment to marketing robotics (Rosen, 1999). However, one of the greatest challenges in today's industrial robotics is still the mismatch between control models and the physical robots, making the so desired robot off-line programming with accurate positioning largely used in industry, an achievement quite far away to be reached. That means, robots have a very good repeatability, but still a poor accuracy (Lightcap et al., 2008).

Off-line programming is, by definition, the technique of generating a robot program without using the real robot and offers many advantages over the on-line method (Motta, 2007). However, there are several obstacles for making off-line programming viable. One of those obstacles is the lack of accuracy in static positioning of robots, and that is where robot calibration plays an important role.

In addition to improving robot accuracy through software (rather than by changing the mechanical structure or design of the robot), calibration techniques can also minimize the risk of having to change application programs due to slight changes or drifts (wearing of parts, dimension drifts or tolerances, and component replacement effects) in the robot system. This is mostly important in applications that may involve a large number of task points.

Robot calibration is an integrated process of modeling, measurement, numeric identification of actual physical characteristics of a robot, and implementation of a new model (Schröer, 1993; Motta, 2007). The proposal of this article is to present a robot calibration system that has been developed aiming at improving robot position accuracy and to present the main theoretical and practical aspects to consider when building robot calibration systems for off-line programming, including industrial robots and robots specially designed for specific tasks. Mathematical basics, experimental procedures and results are presented and discussed. The system was conceived to be used with an ABB IRB2000 robot model; however, it can be easily adapted and used with any type of industrial robots or robots specifically designed for non-standard applications.

The Robot Calibration System

Robot calibration is the process of improving the robot accuracy by modifying its control software (Bernhardt and Albright, 1993). General calibration systems can be divided into two main groups: static and dynamic (Schröer, 1993). Static calibration is an identification of those parameters which influence mainly static positioning characteristics of a robot (position and orientation of the end-effector), while dynamic calibration is used to identify parameters influencing primarily motion characteristics (velocity and forces). Static calibration systems focus mainly on the correction of geometrical parameters such as joint-axis geometries and joint angle off-sets. Non-geometric parameters include compliance (joint and link elasticity), gear form errors (eccentricity and transmission errors), gear backlash, and temperature related expansion. Both geometrical and non-geometrical parameters are included in static robot calibration modeling, since parameters can be measured from the robot poses only. Once robot's static parameters are identified, a dynamic calibration can take place. This type of calibration is performed to determine dynamic related characteristics of the robot (e.g. distribution of mass in the links, friction in actuators and joints, stiffness, etc.). Internal characteristics such as friction tend to be difficult to identify accurately due to their coupling with other dynamic parameters. Dynamic robot calibration takes importance only in large robots subject to high velocities and accelerations and needs very cumbersome experimental procedures (Raucent and Samin, 1993).

Apart from the fact that there have already been quite a few publications (Chen and Chao, 1987; Duelen and Schröer, 1991; Vincze et al., 1999) concerning non-geometrical parameter calibration, these extra parameters come with a high cost in terms of model complexity. Schröer (1993) reported that the significant or relevant kinds of parameters have the following order of importance: geometric-kinematic parameters, joint elasticity and link elasticity. Transmission and coupling (i.e. gear parameters) are insignificant for improvement of pose accuracy. In this work only static calibration with geometrical errors will be considered, since they are the main source (≈90%) of the total position errors in industrial robots (Stark, Benz and Hüttenhofer, 1993). More recent publications concerning the extension of robot calibration

Paper accepted February, 2010. Technical Editor: Glauco Caurin

approaches to articulated arm coordinate measuring machines (AACMM) also considers only geometrical parameters for the same reasons (Santolaria et al., 2008).

The calibration system described here involves the joint work of a measurement system, an off-line robot calibration model and the robot controller as shown in Fig. 1.

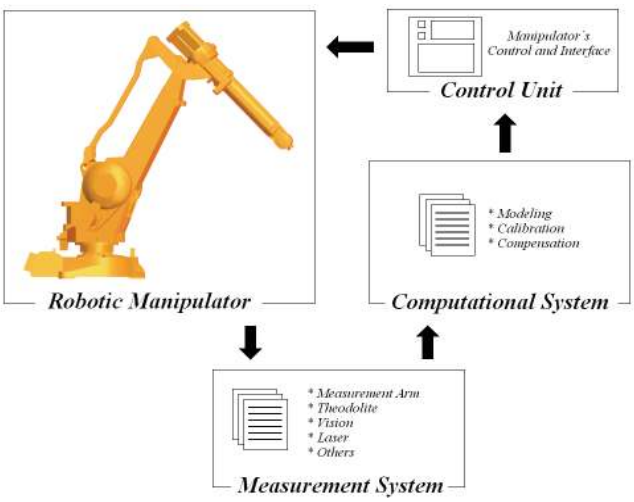

Figure 1. Block Diagram of the robot calibration system.

The blocks in Fig. 1 show a robotic manipulator which has its poses measured by any type of measurement system that sends the robot end-effector coordinates to an off-line computer system, represented in any external coordinate frame. The computer system includes all the mathematics involved with modeling error parameters, parameter identification routines and procedures to compensate robot joint coordinates. New robot joint coordinates are output to the robot control unit to move the robot to a pose closer to the target than before the calibration, reducing position errors.

In the next sections, each part of the calibration system will be discussed such that the entire system can be fully understood. The robot calibration model outputs geometric parameters that describe the robot geometry (links lengths, joint offsets and axis misalignments). The robot calibration procedures can be divided into four main steps: 1) kinematic modeling; 2) position measurements; 3) parameter identification; 4) position compensation.

Kinematic modeling is a subject that has been widely studied for a long time, and together with dynamics it is the topic in robotics that has produced the largest number of publications up to date (Goldenberg and Emami, 1999). Kinematic modeling for robot calibration has to include an error model to fit the actual robot errors.

The measurement step is the most critical in the shop-floor, since measurement data have to be many times more accurate than the robot's accuracy expected after the calibration procedures. There is a wide range of measurement systems available with different levels of accuracy (Kyle, 1993; Hidalgo and Brunn, 1998), including contact and non-contact systems, from theodolites to laser systems, vision-based systems, ultra-sound and coordinate measuring machines, with several price ranges and accuracy. The measurement system adopted for this work can only measure end-effector positions, since orientation measuring is not possible with the contact probe-based type of measuring device used. Only few measuring systems have this capacity and some of them are usually based on vision or optical devices. The measurement system used is a Measurement Arm that has to be manually moved to the targeted position with a contact probe. Details are shown in section 2.2. Vision-based measurement systems designed for robot calibration are cheap and can be used within large measuring volumes with a reasonable accuracy (Motta et al., 2001). The price of the measuring system appears to be a very important issue for medium size or small companies.

Parameter identification is the step where data acquired with the measurement system are processed by using a mathematical model specific for error searching, producing a corrected robot kinematic model. The errors calculated are used to fit the robot model to the experimental data.

Position compensation refers to using the robot geometrical errors calculated from the parameter identification step in the robot kinematic model to modify the robot's control commands, compensating joint positions as needed to improve the robot position accuracy.

Kinematic Modeling

The first step to calibrate a robotic manipulator is kinematic modeling. The IRB2000 robot is an industrial robot with six degrees of freedom used in a wide range of tasks, from welding to palletizing and spray-painting (ABB, 1993).

A robot can be seen as a series of links which connects its end-effector to its base, with each link connected to the next by an actuated joint (McKerrow, 1991). The kinematic model describes mathematically those links and joints.

There are many desirable characteristics for a kinematic model, but when considering kinematic models constructed aiming at using in robot calibration procedures three are mostly important: completeness, continuity and minimality (Motta, 2005; Albright, 1993). Completeness is the ability of a kinematic model to describe all possible spatial geometric joint configurations of a robot. Continuity and minimality influence directly robot calibration, since they are related to model smoothness and to parameter redundancies in the model, respectively.

Robot kinematic models are generally based on the Denavit-Hartenberg convention (McKerrow, 1991) because of its simplicity and easiness to be geometrically represented. The elementary transformations can be formulated as (Denavit-Hartemberg convention):

$$T = f(\theta, \alpha, d, l) = R_Z(\theta) T_Z(d) T_X(l) R_X(\alpha) \tag{1}$$

where T represents position and orientation coordinates of a link frame related to a previous one, where θ and α are the rotation parameters, d and l are translation parameters.

However, when considering an error parameter model for robot calibration a 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. An 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 $p \in \mathrm{R}^6$ (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. Details about the Jacobian matrix will be discussed ahead.

The IRB2000 robot (Fig. 2 and Fig. 3) has perpendicular and parallel axes. However, the Denavit-Hartemberg convention, shown in Eq. (1), cannot be used in error parameter models when modeling parallel axes due to singularities that occur in the Jacobian matrix, as explained. This matrix will be described ahead in the text in Eq. (5). This issue is discussed in details in Motta (2005) and Schröer (1997). A possible convention for parallel axes is the Hayati-Mirmirani (1985) that cannot be used in perpendicular axes for the same reason. The Hayati-Mirmirani is a four-parameter convention that describes the transformation between two parallel axes as shown in Eq. (2) (Hayati-Mirmirani convention):

$$f(\theta,\alpha,\beta,l) = R_Z(\theta)T_X(l)R_X(\alpha)R_Y(\beta) \tag{2}$$

Each joint coordinate system here is orthogonal, and the axes obey the right-hand rule. In Fig. 2 the base coordinate frame (b) (robot reference) is assigned with axes parallel to the world coordinate frame (w). 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.

For revolute joints the zero position is taken as the one with all x axes of the link coordinate frames parallel or with the same direction. The z-axes are coincident with the joint axes. Coordinate frames do not move relative to the link it is attached to, and the succeeding link moves relative to it. Coordinate frame i refers to joint $i+1$, that is, the joint that connects link i to link $i+1$.

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 Tn_i is the distance between coordinate frames $i-1$ and i, and n is the parallel axis in the coordinate system $i-1$. Figure 2 shows the above rules applied to the IRB-2000 robot with all the coordinate frames and geometric features.

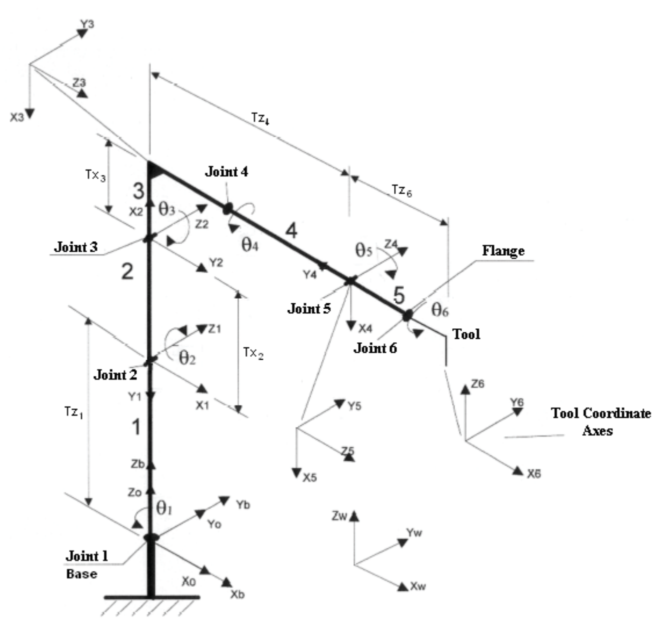

Figure 2. Kinematic Model of the IRB-2000 robot.

Table 1. Initial Values of Model Parameters of Links and Joints of the IRB2000 Robot (units in mm and degrees): (V) – Model Variable, (M) – Parameter Value.

V	M	V	M
Link B		**Link 0**	
$T_{x_b} + \delta_{Tx_b}$	0.00	T_{x_0}	0.00
$T_{y_b} + \delta_{Ty_b}$	0.00	R_{z_0}	0.00
$T_{z_b} + \delta_{Tz_b}$	0.00	T_{z_0}	0.00
$R_{x_b} + \delta_{Rx_b}$	0.00	R_{x_0}	0.00
$R_{y_b} + \delta_{Ry_b}$	0.00		
$R_{z_b} + \delta_{Rz_b}$	0.00		
Link 1		**Link 2**	
$T_{x_1} + \delta_{Tx_1}$	0.00	$T_{x_2} + \delta_{Tx_2}$	710.00
$T_{z_1} + \delta_{Tz_1}$	750.00	$R_{y_2} + \delta_{Ry_2}$	0
$R_{x_1} + \delta_{Rx_1}$	-90.00	$R_{x_2} + \delta_{Rx_2}$	0
R_{z_1}	0.00	$R_{z_2} + \delta_{Rz_2}$	-90.00
Link 3		**Link 4**	
$T_{x_3} + \delta_{Tx_3}$	-125.00	T_{x_4}	0.00
$T_{z_3} + \delta_{Tz_3}$	0.00	$T_{z_4} + \delta_{Tz_4}$	850.00
$R_{x_3} + \delta_{Rx_3}$	90.00	$R_{x_4} + \delta_{Rx_4}$	-90.00
$R_{z_3} + \delta_{Rz_3}$	180.00	$R_{z_4} + \delta_{Rz_4}$	0.00
Link 5		**Link 6**	
T_{x_5}	0.00	$T_{x_6} + \delta_{Tx_6}$	0.00
T_{z_5}	0.00	$T_{y_6} + \delta_{Ty_6}$	0.00
$R_{x_5} + \delta_{Rx_5}$	90.00	$T_{z_6} + \delta_{Tz_6}$	100.00
$R_{z_5} + \delta_{Rz_5}$	0.00	R_{x_6}	0.00
		R_{y_6}	-90.00
		R_{z_6}	0.00

Using the previous two conventions (Denavit-Hartemberg and Hayati-Mirmirani) and taking into account the requirements of a kinematic model (completeness, continuity and minimality), the singularity-free approach discussed was applied for the assignment of coordinate frames and for the definition of which error parameters should be included in the kinematic model (Motta, 2005). Using this approach and the mechanical drawings of the IRB2000 (ABB, 1993), a kinematic model representing mathematically this robot was constructed. The parameters used are shown in Table 1, where δ are the error parameters between the nominal model and the actual robot model to be identified by the calibration system, and are initially set to null. The Hayati-Mirmirani convention was used to relate joints 2 and 3 (parallel) and Denavit-Hartemberg the other joints (perpendicular).

The correct choice of the error parameters are of vital importance to the minimality and continuity of the kinematic model and, as discussed by Motta (2005), the error parameters are included in the model at links in such a way that there will be no redundancies. A discussion about the choice of those parameters and about strategies to analyze the conditioning of the resultant system is shown in Motta and McMaster (1999).

Figure 3. IRB2000 and its 3D representation.

The base link (Link B) and the last link (Link 6, related to the TCP), shown in Table 1, are the transformations that locate the robot base frame related to the measurement system coordinate frame and the tool center position and orientation related to the robot flange respectively. As both coordinate systems may vary in position and orientation and cannot be measured, it is required that their elementary transformations be a Euclidean group of parameters, with 6 parameters each.

Measurement System

The robot calibration computer system constructed gives support to different measurement systems, thanks to its modular construction. In this work, the measurement system used was a Measurement Arm ITG ROMER with an accuracy reported by the manufacturer of 0.087 mm. The system can be seen in Fig. 4. The measurement arm was used to measure the end-effector positions of the ABB IRB2000 robot.

Figure 4. ITG ROMER measurement arm.

Mathematical Basis for Parameter Identification

Concerning mathematics, robot calibration is basically a problem of fitting a non-linear model to experimental data. The results are error parameters that are identified using a proper cost function.

A robot kinematic model can be seen as a function that relates kinematic model parameters and joint variables to coordinate positions of the robot end-effector. As an example to present the mathematics involved, one can define $P = T_1 \,.\, T_2 \,. \,.. T_m$, where P is the manipulator transformation, T_i is each of the link transformations defined in Eq. (1) and m is the number of links. Thus, a kinematic model following the Denavit-Hartenberg convention can be derived as (from Eq. (1)):

$$\Delta P = \frac{\partial P}{\partial \theta}\Delta\theta + \frac{\partial P}{\partial \alpha}\Delta\alpha + \frac{\partial P}{\partial d}\Delta d + \frac{\partial P}{\partial l}\Delta l \qquad (3)$$

where P represents position and orientation coordinates of the manipulator end-effector (Tool Center Position – TCP) and θ , α , d and l are the four parameters that define the transformation from a robot joint frame to the next joint frame, where θ and α are the rotation parameters, d and l are translation parameters.

The first derivative shown in Eq. (3) can be interpreted as the position and orientation error equation of the robot TCP coordinates (Hollerbach and Benett, 1988), where ΔP is the pose error and can be physically measured. Considering the manipulator transformation, P, from the robot's base frame to the TCP-frame, the measured robot position, M, related to the measurement system coordinate frame and the transformation that locates the robot base frame to the measurement system, B, then ΔP is the vector illustrated in Fig. 5.

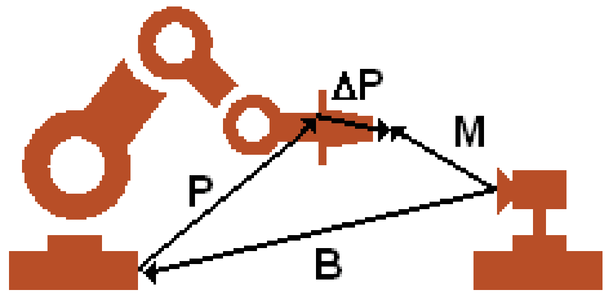

Figure 5. Calibration transformations.

The transformation B can also be considered as a link that makes part of the robot model in such a way that Fig. 5 can be presented as in Fig. 6. Then the error value ΔP can be calculated using Eq. (4).

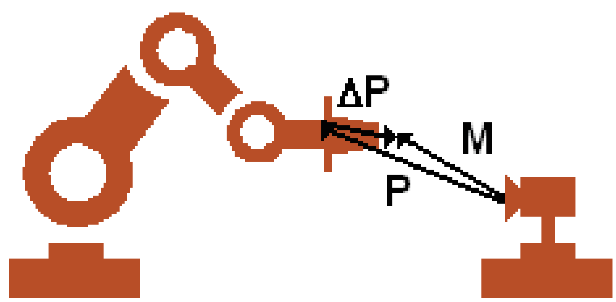

Figure 6. Simplified representation of the calibration transformations.

$$\Delta P = M - P \qquad (4)$$

The transformation P is then iteratively modified when the error parameters of the robot model are updated, and by the end of the calibration process, the transformation P represents the actual robot and its location in the measurement system coordinate frame.

Rewriting Eq. (3) in a matricial form for various measured positions and orientations of the robot end-effector, Eq. (4) can be formulated as the Jacobian matrix containing the partial derivatives of P such that Δx is the vector of the model parameter errors:

$$\begin{bmatrix} \Delta P_1 \\ \Delta P_2 \\ \vdots \\ \Delta P_n \end{bmatrix} = \begin{bmatrix} \partial P_1/\partial\theta & \partial P_1/\partial\alpha & \partial P_1/\partial d & \partial P_1/\partial l \\ \partial P_2/\partial\theta & \partial P_2/\partial\alpha & \partial P_2/\partial d & \partial P_2/\partial l \\ \vdots & \vdots & \vdots & \vdots \\ \partial P_n/\partial\theta & \partial P_n/\partial\alpha & \partial P_n/\partial d & \partial P_n/\partial l \end{bmatrix} \begin{bmatrix} \Delta\theta \\ \Delta\alpha \\ \Delta d \\ \Delta l \end{bmatrix} = \begin{bmatrix} J_1 \\ J_2 \\ \vdots \\ J_n \end{bmatrix} \Delta x \Rightarrow \mathbf{J}\Delta x = \mathbf{\Delta P} \tag{5}$$

The size of the Jacobian depends on the number of measured points selected to be measured in the workspace (m) and the number of error parameters included in the model (n) (matrix order = m x n). Thus the calibration problem is reduced to the solution of a non-linear system of the type $\boldsymbol{A}x = \boldsymbol{b}$.

There are many different methods to solve this type of system and one which is widely used is the Squared Sum Minimization (SSM). Many authors (Jacoby et al., 1972; Dennis and Schnabel, 1983) discuss extensively those methods and algorithms are easily found in the literature (Press et al., 1992).

One method to solve non-linear least-square problems that proved to be very successful in practice and thus recommended for general solutions is the algorithm proposed by Levenberg-Marquardt (LM algorithm) (Dennis and Schnabel, 1983). Several algorithms versions of the L.M. algorithm have been proved to be successful (globally convergent). It turns to be an iterative solution method by introducing few modifications in the Gauss-Newton method in order to overcome some divergence problems (Jacoby et al., 1972).

Each algorithm iteration has three steps, where x_k represents the parameter list of the mathematical model in the k^{th} iteration and Δx_k the alterations to be introduced in the model (Motta et al., 2001).

1. Calculation of the robot's Jacobian ($\mathbf{J}(x_k)$)
2. Calculation of the vector Δx_k using the relation
   $\Delta x_k = -\{[\mathbf{J}(x_k)]^T \mathbf{J}(x_k) + \mu_k \mathbf{I}\}^{-1} [\mathbf{J}(x_k)]^T \mathbf{\Delta} P(x_k)$
3. Update $x_{k+1} = x_k + \Delta x_k$ and $k = k+1$

where μ_k is obtained from the formation law in Eq. (6).

$$\begin{cases} \mu_0 = 0.001 \\ \mu_{k+1} = \begin{cases} 0.001\lambda & if \left\|\mathbf{\Delta}P(x_{k+1})\right\| \geq \left\|\mathbf{\Delta}P(x_k)\right\| \\ 0.001/\lambda & if \left\|\mathbf{\Delta}P(x_{k+1})\right\| < \left\|\mathbf{\Delta}P(x_k)\right\| \end{cases} \\ 2.5 < \lambda < 10 \end{cases} \tag{6}$$

Position Compensation

Since the robot parameter errors are identified, they can be used to predict the robot end-effector pose errors and compensate for such errors. Many techniques for position compensation can be found in the literature (Zhuang and Roth, 1996), among which one of the simplest is the so called Pose Redefinition Method.

The Pose Redefinition Method uses a linearized accuracy error model and predicted position errors to calculate a compensated pose (P_c) to move the robot to the desired robot pose (P_d), according to Eq. (7).

$$P_c = P_d\left[T_x(-d_x)T_y(-d_y)T_z(-d_z)R_x(-\phi_x)R_y(-\phi_y)R_z(-\phi_z)\right] \tag{7}$$

where $[\mathrm{d_x} \quad \mathrm{d_y} \quad \mathrm{d_z} \quad \phi_\mathrm{x} \quad \phi_\mathrm{y} \quad \phi_\mathrm{z}]^\mathrm{T}$ are the pose errors predicted by using the calibrated model, such as d is the position error vector and ϕ is the orientation error vector.

Experimental Procedures

The robot calibration system implemented was evaluated on an ABB IRB2000 robot using the ITG ROMER measurement arm. The robot was calibrated within different workspaces, and the accuracy improvement could be assessed in various robot configurations. The system was also used to validate the correct matching between the nominal robot kinematic model in the off-line calibration software and the nominal robot kinematic model in the control unit. The results and procedures are presented and discussed to show up the performance of the developed system and the robot accuracy improvements.

The IRB2000 Remote Control

Robot calibration procedures require the access to the robot TCP position coordinates and the correspondent joint values. However, the IRB2000 control unit does not show this information on the teach-pendant screen (few industrial robots will do so), but only when expensive off-line programming software produced by the manufacturer is available. Fortunately, the IRB2000 has a remote control interface that complies with those requirements (ABB, 1993). Thus, by using the remote control interface, software for remote manipulation of the IRB2000 was developed. The computer program communicates with the robot using the ADPL10 and ARAP protocols and the RS232 interface. With the help of this software the robot can be commanded to any position within its workspace and joint variables can be read from its control unit and recorded together with TCP coordinates. The software interface can be seen in Fig. 7.

Control Unit Model Identification

When a robot is to be calibrated for the first time, it is very important to check if the kinematic model of the off-line program is exactly the same as the robot control unit, since operation manuals from manufacturers do not always inform precise geometric parameter values. Doing so, the kinematic model used in the off-line program can be corrected to fit exactly the nominal model used by the control unit. This procedure is needed because the nominal model used by the control unit is not accessible.

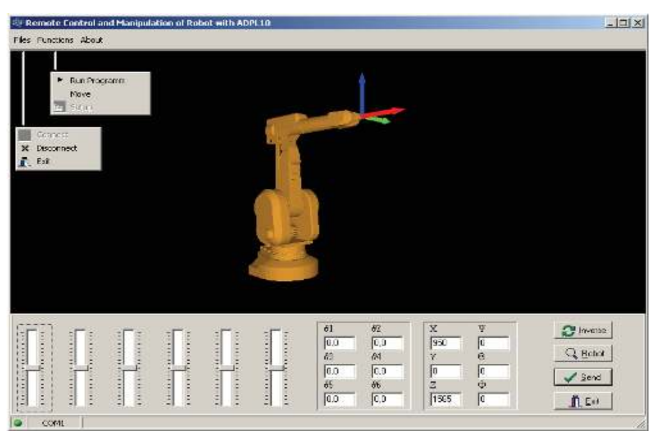

Figure 7. Remote control software.

The procedure to identify the control unit kinematic model requires the robot to be moved to several positions within the workspace and the joint variables and TCP positions to be recorded. Then, the value of ΔP, in Eq. (4) (Fig. 5), can be fully determined, where B is set to null and M is the TCP positions obtained from the control unit. The error parameters in the kinematic model are then modified to include only error variables related to the link dimensions in the nominal kinematic model, which means that not all error parameters in Table 1 are identified. Table 2 shows the identified link parameters.

Those results, as expected, do not represent a considerable change in the robot model, and can be considered as numerical error due to the low resolution of the TCP position obtained from the control unit (0.125 mm). Therefore, it is not needed that those results are incorporated into the nominal model of the robot IRB 2000. However, in some robots, like the PUMA-500, these authors found significant mismatches between the robot geometrical parameters printed in manufacturer manual and the ones found in the experimental tests. In modular robots, constructed specifically for certain tasks, the nominal parameters can never be accurately determined without these procedures.

Table 2. Identified error parameters for the IRB2000 controller model (units in mm).

Error Parameter	Value
δ_{Tz_1}	0.08
δ_{Tx_2}	0.20
δ_{Tx_3}	-0.05
δ_{Tz_4}	0.05
δ_{Tz_6}	0.08

Robot Calibration Results

With the mathematical model used by the robot's control unit identified, the next step is the identification of the mathematical model that best represents the actual robot. At this stage the robot is maneuvered to different positions, and those positions are measured using the measurement system. The value of each joint variable is obtained from the control unit and the position of the TCP is measured using the measurement system. So, the values of all vectors, shown in Eq. (4) and Fig. 6, are known and are fully determined.

For the calibration of the IRB2000 model shown in Table 1, different workspace volumes and calibration points were selected, aiming at spanning from large to smaller regions. Five calibration volumes were chosen within the robot workspace. The volumes were cubic shaped. In Fig. 8 and Fig. 9 it is shown the calibration points distributed on the cubic faces of the calibration volumes. The two external cubes have 12 calibration points (600 mm) and the three internal cubes (600, 400 and 200 mm) have 27 positions.

The experimental routine was ordered in the following sequence: 1) robot positioning; 2) robot joint positions recorded from the robot controller (an interface between the robot controller and an external computer has to be available); and 3) robot positions recorded with the external measuring system. In this experiment only TCP positions were measured, since orientation measuring is not possible with the type of measuring device used. Only few measuring systems have this capacity and some of them are usually based on vision or optical devices.

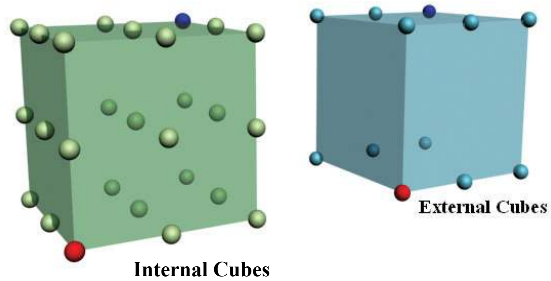

Figure 8. Cubic regions chosen and robot positions within its workspace.

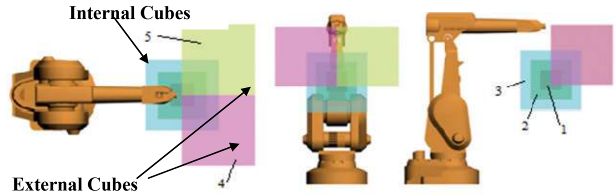

Figure 9. Calibration regions within robot's workspace.

In Fig. 10 the accuracy improvement obtained through the robot calibration is shown. The errors in Fig. 10 were calculated by Eq. (8) as

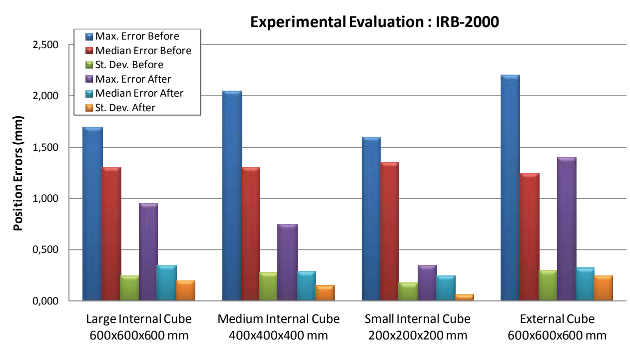

Figure 10. Evaluation of the calibrated model in each of the calibration regions.

$$\begin{cases} \overline{E} = \dfrac{\sum_{i=1}^{n} \left|\Delta \hat{T}_i\right|}{n} = \dfrac{\sum_{i=1}^{n} \left|M_i - P_i\right|}{n} \\ \sigma = \sqrt{\dfrac{\sum_{i=1}^{n} \left(\left|M_i - P_i\right| - \overline{E}\right)^2}{n}} \end{cases} \tag{8}$$

according to the notation of Eq.(4).

The results shown pinpoint the importance of robot calibration where robot accuracy is an important issue in off-line programmed tasks. By performing the calibration of the IRB2000 robot, position errors were reduced from above 1.4 mm to below 0.3 mm.

Every time a robot moves from a region of the workspace to another, the robot base has to be recalibrated, since the errors calculated from the parameter identification routine include the robot base geometrical parameters, which cannot be measured. However, with an off-line programmed robot, with or without calibration, that has to be done anyway. In a similar way, if the robot tool has to be replaced, or after an accident damaging it, it is not necessary to recalibrate the entire robot, only the tool. For that, all that has to be done is to place the tool at few physical marks with known world coordinates (if only the tool is to be calibrated not more than six) and run the off-line calibration system to find the actual tool coordinates represented in the robot base frame.

Figure 10 shows that the accuracy was improved approximately by the same amount either within the internal calibration cube or within the external ones (same sizes). It also shows that the larger the work volume is, the achieved accuracy improvement is slightly smaller. However, with larger calibration volumes it is possible to move the robot joint angles in a broader range, which means the robot will be better calibrated in a larger volume of the workspace, even when working outside that volume. The limitation in the size of the calibration volume appears to be the capacity of the measurement system to measure accurately in a broader space.

Conclusions

A robot calibration system capable of improving the robot position accuracy has been presented. The development of this system with its mathematical background and practical considerations focusing on the IRB2000 robot were discussed and presented together with a brief review on kinematic modeling and the major role it represents in robot calibration systems.

The system was tested by using an ITG ROMER measurement arm, and results showed an average improvement of the robot accuracy from 1.4 to 0.3 mm. When using a minimal but complete set of model parameters and proper modeling conventions to avoid Jacobian singularities in the parameter identification routines, the robot calibration computer system presented here offers a faster algorithm convergence to the optimal solution of variable parameters and needs less number of measurement points for that than systems that do not follow the same mathematical optimization scheme.

The approach used to construct robot models is ready to be used in a friendly work environment for calibration purposes with the flexibility needed to work on a wide range of industrial robots regardless the manufacturer or robot geometry. It is a tool aiming at making the implementation of off-line programming on the shop-floor of industrial robot applications a more viable procedure.

The work presented includes simple and feasible methodology to build robot calibration systems, specially oriented to calibrate robots any time there is a change in the robot links, such as changing parts or tools, using cooperative robots, specially designed robots or whenever it is important to know the actual kinematic parameters of a robot model.

It was also presented and discussed a method to find the nominal kinematic model parameters of a robot, which is essential when modeling off-line programming applications and that information is not available from commercial industrial robots.

Acknowledgements

The authors acknowledge the support provided by the Brazilian Research Council, CNPq.

References

ABB, 1993, Robotics Products: IRB2000.

Albright, S.L., 1993, "Calibration System for Robot Production Control and Accuracy", In: "Robot Calibration", Bernhardt and Albright, Chapman & Hall, ISBN 0 412 49140 0, Cambridge, UK, pp. 37-56.

Bernhardt, R., Albright, S.L., 1993, Introduction, "Robot Calibration", Bernhardt and Albright, Chapman & Hall, ISBN 0 412 49140 0, Cambridge, UK, pp. 3-11.

Chen, J. and Chao, L.M., 1987, "Positioning Error Analysis for Robot Manipulators with all Rotary Joints", *IEEE Journal of Robotics and Automation*, Vol. RA-3, No. 6, pp. 539-545.

Dennis, J.E., Schnabel, R.B., 1983, "Numerical Methods for Unconstrained Optimization and Nonlinear Equations", Prentice Hall, New Jersey, USA.

Duelen, G. and Schröer, K., 1991, "Robot Calibration-Methods and Results", *Robotics & Computer-Integrated Manufacturing*, Vol. 8, No. 4, pp. 223-231.

Goldenberg, A.A., Emami, M.R. , 1999, "Kinematics and Dynamics of Robot Manipulators", In: Handbook of Industrial Robotics, Nof, S. Y., 2nd ed., John Wiley & Sons, ISBN 0-471-17783-0, USA, pp. 79-98.

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.

Hidalgo, F. and Brunn, P., 1998, "Robot Metrology and Calibration Systems – A Market Review", *Industrial Robot*, Vol. 25, No. 1, pp. 42-47.

Hollerbach, J.M., Benett, D.J., 1988, "A Survey of Kinematic Calibration", *Robotics Review*, pp. 207-242.

Jacoby, S.L.S, Kowalik, J.S., Pizzo, J.T., 1972, "Iterative Methods for Nonlinear Optimization Problems", Ed. Prentice Hall, New Jersey, USA.

Kyle, S.A., 1993, "Calibration System for Robot Production Control and Accuracy", In: "Robot Calibration", Bernhardt and Albright, Chapman & Hall, ISBN 0 412 49140 0, Cambridge, UK, pp. 79-100.

Lightcap, C., Hamner, S., Schmitz, T. and Banks, S., 2008, "Improved Positioning Accuracy of the PA10-6CE Robot with Geometric and Flexibility Calibration", *IEEE Transactions on Robotics*, Vol. 24, No. 2, pp. 452-456.

McKerrow, P.J., 1991, "Introduction to Robotics", Addison-Wesley, Singapore.

Motta, J.M.S.T. and McMaster, R.S., 1999, "Modeling, Optimizing and Simulating Robot Calibration with Accuracy Improvement", *Journal of the Brazilian Society of Mechanical Sciences and Engineering,* Vol. 21, No. 3, pp. 386-402.

Motta, J.M.S.T., 2005, "An Investigation of Singularities in Robot Kinematic Chains Aiming at Building Robot Calibration Models for Off-line Programming", *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, Vol. 27, No. 2, pp. 200-204.

Motta, J.M.S.T., 2007, "Robot Calibration: Modeling, Measurement and Applications", Industrial Robotics: Programming, Simulation and Applications, Low Kin-Huat. (Org.), 1st ed. Mammendorf – Germany: Verlag Robert Mayer-Scholz, Vol. 1, pp. 107-130.

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*. Elsevier Science, Vol. 17, No. 6, pp. 487-49.

Press, W.H., Teukolsky, S.A., Vetterling, S.A., Flannery, B.P., 1992, "Numerical Recipes in C: The Art of Scientific Computing", Cambridge University Press, Cambridge, UK.

Raucent, B. and Samin, J.C., 1993, "Modelling and Identifying the Dynamic Parameters of Robot Manipulators", In: "Robot Calibration", Bernhardt and Albright, Chapman & Hall, London, pp. 197-232.

Rosen, C.A., 1999, "Robots and Machine Inteligence", In: Handbook of Industrial Robotics, Nof, S. Y., 2nd ed., pp. 19-30, John Wiley & Sons, ISBN 0-471-17783-0, USA.

Santolaria, J., Aguilar, J.-J., Yague, J.-A. and Pastor, J., 2008, "Kinematic parameter estimation technique for calibration and repeatability improvement of articulated arm coordinate measuring machines", *Precision Engineering*, Vol. 32, pp. 251-268.

Schröer, K., 1993, "Theory of Kinematic Modeling and Numerical Procedures for Robot Calibration", In: "Robot Calibration", Bernhardt and Albright, Chapman & Hall, ISBN 0 412 49140 0, Cambridge, UK, pp. 157-193.

Schröer, K., 1997, "Complete, Minimal and Model-Continuous Kinematic Models for Robot Calibration", *Robotics & Computer-Integrated Manufacturing*, Vol. 13, No. 1, pp. 73-85.

Stark, G., Benz, E. and Hüttenhofer, M., 1993, "Calibration Experiences in Industry", In: Robot Calibration, Bernhardt and Albright, Chapman & Hall, ISBN 0 412 49140 0, Cambridge, UK, pp. 283-295.

Vincze M, Spiess S, Parotidis M, Götz M., 1999, "Automatic generation of nonredundant and complete models for geometric and non-geometric errors of robots", *International Journal of Modelling and Simulation*, Vol. 19, No. 3, pp. 236-243.

Zhuang, H., Roth, Z.S., 1996, Camera-Aided Robot Calibration, Ed. CRC Press, USA.