Inverse kinematics robot calibration by spline functions

A. Doria and F. Angrilli

Università di Padova, Dip. di Ingegneria Meccanica, Padova, Italy

S. De Marchi

Università di Padova, Dip. di Matematica Pura e Applicata, Padova, Italy

Inverse kinematics calibration is a technique used to improve the accuracy of robot manipulators when they are programmed off-line. This technique requires an approximation of position deviations of the end-effector of the robot. In this paper three-variate splines are used as approximating functions; a stable and efficient algorithm for their computation is presented. Many experimental tests are performed to evaluate the merits of the approximation. The improvements in the accuracy of the robot, obtained after calibration, demonstrate the validity of the method.

Keywords: robot, calibration, approximation, splines

1. Introduction

There are many applications in engineering in which the availability of a suitable approximation algorithm is required; one of these is *robot calibration*. The operation of a robot manipulator is usually specified in terms of a succession of *poses* of the end-effector. If a teaching method is used, an operator leads the robot to the desired poses and records in the memory of the robot the corresponding values of joint displacements. The sequence is then repeated automatically and the end-effector pose is affected only by the repeatability deviations that are of stochastic nature. If an off-line programming technique is used, the succession of desired poses is specified by means of a computer. When a robot is programmed off-line, joint displacements are calculated by performing inverse kinematic analysis of the nominal model of the robot. The nominal model usually differs from the actual model of the manipulator because of manufacturing errors, clearances, joint compliances, etc.; therefore joint displacements calculated by means of the nominal model are not correct and produce deviations in the end-effector pose that add to the stochastic deviations. Therefore a calibration technique is required to increase the manipulator accuracy when a robot manipulator is programmed off-line.

Some authors developed so-called *forward calibration* (see Refs. 1–3). The aim of their work was to obtain a more accurate model of the robot by identifying the actual parameters of the robot on the basis of measurements in the workspace. This approach can give good results but may be difficult, and some problems arise if an inverse model of the robot is required. In fact, for the "nominal robot" the inverse solution is usually available because it is kinematically simple (some axes intersect or are parallel) but for the "actual robot" with axis misalignments, many offsets, and nongeometric errors, sometimes only a numerical solution can be found.

To cope with these problems, the "*inverse calibration*" approach was proposed by some authors (see, for example, Refs. 4 and 5). With this approach no attempt to modify the robot model is made, but some *correction functions* are calculated. If the corrections are added to the cartesian parameters that define the pose, the calibration is called *precorrective*. If the corrections are added to the nominal joint angles that would produce the desired pose, the inverse calibration is called *postcorrective*. The correction functions are calculated by evaluating the position and orientation deviations in some poses in the workspace. When pose $\vec{x}$ is implemented in the computer, joint displacements $\vec{\theta}$ are calculated and the end-effector reaches pose $\vec{x}'$. Then, the measured pose deviation is $\vec{E}_p = \vec{x}' - \vec{x}$. If pose $\vec{x}'$ is commanded, the correction $-\vec{E}_p$ must be implemented. Indeed on the basis of $\vec{x}' - \vec{E}_p = \vec{x}$, the computer calculates joint displacements that locate the end-effector in the desired pose. Thus, if a precorrective technique is used, correction vector

Address reprint requests to Prof. Angrilli at the Università di Padova, Dip. di Ingegneria Meccanica, Via Venezia, I-35131 Padova, Italy.

Received 13 February 1992; revised 3 December 1992; accepted 4 December 1992

$-\vec{E}_p$ must be calculated in all the locations of the workspace. Since it is impossible to carry out all the measurements, the vector $-\vec{E}_p$ is measured only in some positions of the workspace (these positions are called *training points*) and the corrections in the other poses are calculated by means of approximation functions.

In previous works, linear functions (see Ref. 4) and third-order three-variate polynomials (see Ref. 5) were proposed and tested as correction functions.

The aim of this paper is to contribute to the development of pre-corrective inverse calibration by testing the effectiveness of approximation with multivariate spline functions.

The paper is subdivided in two main parts: Section 2 concerns approximation with spline functions; section 3 deals with the application of approximating splines to a robot calibration problem.

2. Approximation by spline functions

In this section, we give the basic ideas that we have followed in choosing the class of spline functions as the set of our approximating functions. Our problem can be formulated as follows:

Problem 1. *Which class of functions is eligible to be correction functions?* From the literature it is known that the class of polynomials is generally suitable on *small* intervals. Moreover, polynomials tend to show oscillations close to the limits of the interval. This tendency to oscillate becomes increasingly pronounced as the order of the polynomial is increased (see Ref. 6). Another limitation of the approximation by polynomials is the *global definition*. This means that if we change a point in the set of the points chosen as knots, all the behavior of the approximating polynomial will be changed. This obviously leads to increased computation.

The above considerations imply that it is better to find a class of low-order piecewise approximating polynomials. *Splines* are piecewise polynomials that are more flexible and hence can fit general functions or data (for a prescribed number of parameters) better than polynomials. These characteristics apply to both interpolation and approximation.

2.1 Some remarks on spline functions

In one dimension a spline function of degree $n - 1$ (or order n) is a piecewise polynomial belonging to $C^{n-2}([a,b])$, where $[a,b]$ is the definition interval.[7]

Definition 2.1.1. Given a strictly increasing set of real numbers

$$a = x_1, x_2, \ldots, x_m = b$$

($m \neq n$, generally $m > n$) a spline function $s(x)$ of order n with knots $x_1, \ldots, x_m$ is a function possessing the following two properties:

(1) in each of the intervals

$$x \leq x_1; \quad x_{j-1} \leq x \leq x_j (j = 2, \ldots, m); \quad x_m \leq x$$

$s(x)$ is a polynomial of degree $n - 1$ or less;

(2) $s(x) \in C^{n-2}(R)$.

We will use the notation $S = S(n - 1; x_1, \ldots, x_m)$ for the linear space of spline functions of degree $n - 1$ with sequence of knots $\{x_1, \ldots, x_m\}$. One representation of a spline of degree $n - 1$ based on the strictly increasing set of knots $x_1, \ldots, x_m$ is

$$s(x) = \sum_{j=0}^{n-1} c_j x^j + \sum_{j=1}^{m} d_j (x - x_j)_+^{n-1} \quad x \in [a,b] \tag{1}$$

where $(x - x_j)_+ = max(0, x - x_j)$. The coefficients $\{c_j: j = 0, \ldots, n - 1\}$ and $\{d_j: j = 1, \ldots, m\}$ distinguish different members of S. The dimension of S is $n + m$. A representation much more convenient for practical purposes, still requiring $m + n$ coefficients in its definition, is given by expressing $s(x)$ as B-splines as suggested in Refs. 8 and 9. Then, choosing $\mathrm{B} = \{B_j: j = 1, \ldots, m + n\}$, a basis of linear independent functions of S that we will call B-Splines or BS, we can express $s(x)$ as a linear combination of $\{B_j\}_{j=1,\ldots,m+n}$. To do this, we need to introduce $2n$ additional knots (see Ref. 10)*

$$x_{-n+1}, x_{-n+2}, \ldots, x_0$$

and

$$x_{m+1}, x_{m+2}, \ldots, x_{m+n}$$

where

$$x_{-n+1} < x_{-n+2} < \cdots < x_1 = a$$

and

$$b = x_m < x_{m+1} < x_{m+2} < \cdots < x_{m+n}$$

Then, we can write

$$s(x) = \sum_{j=1}^{m+n} c_j B_j(x) \quad x \in [x_0, x_{m+1}] \tag{2}$$

There are two fundamental algorithms to compute B-splines. The former, based on the recursive definition of *divided differences* proved to be unstable.[7] The second, based on the *recursive relation* between B-splines, was proposed in Refs. 7 and 9. The recursive relation, for the jth B-spline of order $n > 1$ is

$$B_j^{(n)}(x) = \frac{(x - x_{j-n}) B_{j-1}^{(n-1)}(x) + (x_j - x) B_j^{(n-1)}(x)}{x_j - x_{j-n}} \tag{3}$$

Because $B_{j-1}^{(n-1)}(x)$, $B_j^{(n-1)}(x)$, $x - x_{j-n}$, $x_j - x$ and $x_j - x_{j-n}$ are all positive or null, it follows that there will be no cancellation in evaluating the right-hand side of (3). This is why the algorithm that calculates the BS basis using (3) is stable.

Instead of using $B_j^{(n)}(x)$ as in (3), it is more convenient to work with *normalized* B-splines, i.e.

$$N_j^{(n)}(x) = (x_j - x_{j-n}) B_j^{(n)}(x) \quad x \in [a,b] \tag{4}$$

The following algorithm was used to compute all the BS functions, by computing normalized B-splines:

* As suggested in Ref. 11, we can choose the additional knots satisfying the order $\leq$.

Algorithm 1

```
Assume that $x_{i-n} < x < x_i$; otherwise $B_i^{(n)}(x) = 0$
    1. For $j = i - n + 1, i - n + 2, \ldots, i$ do
        $B_j^{(1)}(x) = \begin{cases} (x_j - x_{j-1})^{-1} & \text{if } x_{j-1} < x \leq x_j \\ 0 & \text{otherwise} \end{cases}$
        $B_j^{(1)} = B_j^{(1)}(x)$;
    2. For $r = 2,3, \ldots, n$ do
        For $j = i - n + r, i - n + r + 1, \ldots, i$ do
            $B_j^{(r)} = \dfrac{(x - x_{j-r})B_{j-1}^{(r-1)}(x) + (x_j - x)B_j^{(r-1)}(x)}{x_j - x_{j-r}}$
At the end we have $B_i^{(n)}(x) = B_i^{(n)}$
```

2.2 Multivariate parametric approximating splines

The parametric representation of a bidimensional spline $s(\tau)$, with τ that varies on a generic interval $I \subseteq R$, can be expressed by[12,13]

$$s(\tau) = \psi(\tau; P_0, \ldots, P_m) \tag{5}$$

The relation (5) says that s depends on $m + 2$ parameters: τ and the $m + 1$ two-dimensional points $P_0, P_1, \ldots, P_m$. Usually $\tau \in [0,1]$ with value $\tau = i/m$ on the ith point P_i.

The general expression of a parametric spline based on normalized B-splines is

$$s(\tau) = \sum_{i=0}^{m} P_i B_i^{(n)}(\tau) \tag{6}$$

In this case, a suitable choice of a sequence of knots X $= \{x_0, \ldots, x_k\}$ (with $k \neq m$) is the following:

$$\begin{cases} x_i = 0 & 0 \leq i \leq n - 1 \\ x_i = i - n + 1 & n \leq i \leq m \\ x_i = m - n + 2 & m + 1 \leq i \leq m + n \end{cases} \tag{7}$$

The sequence (7) is called *open uniform sequence* of knots. The adjective "open" indicates that the knot values at the ends are multiple, with multiplicity equal to the order n of the spline, i.e., $x_0 = x_1 = \cdots = x_{n-1} = 0$; $x_{m+1} = x_{m+2} = \cdots = x_{m+n} = m - n + 2$. It was proved (see Ref. 12) that an *open uniform* knots sequence is preferable to an *open nonuniform* knots sequence. Such a choice, which we have adopted, gives some information about the coefficient matrix in the interpolation problem in the one-dimensional case. This matrix contains at most "n" nonzero elements in each row and the column position of the first nonzero element in each row is a nondecreasing function of the row number. By using normalized B-splines we can easily verify that the use of the knots sequence (7) implies that the elements in position (1,1) and (q,q) (with q the number of points to interpolate) are equal to 1.[10]

A known technique to generate multivariate splines on rectangular patches is by tensorial products. We want briefly to describe h-variate spline spaces, $h \geq 1$.

The space of h-variate splines S_h is generated from the tensorial product of BS in each direction. Formally, if the directions are given by the vector

$$\vec{a} = (a_1, a_2, \ldots, a_h)$$

and

$$B_{a_1}, B_{a_2}, \ldots, B_{a_h},$$

are the BS basis along those directions, we write

$$S_h = span\left\{ \bigotimes_{i=1}^{h} B_{a_i} \right\} \tag{8}$$

The problem that we studied consisted of approximating a function that depends on three parameters. So, given $m_x + 1$, $m_y + 1$, and $m_z + 1$ points on each direction and orders n_x, n_y, n_z, the three-variate parametric spline $s \in S_3$ is given by

$$s(\tau_x, \tau_y, \tau_z) = \sum_{i=0}^{m_x} \sum_{j=0}^{m_y} \sum_{k=0}^{m_z} \times P_{ijk} B_i^{(n_x)}(\tau_x) B_i^{(n_y)}(\tau_y) B_k^{(n_z)}(\tau_z) \tag{9}$$

where P_{ijk} is the generic four-dimensional point.

The algorithm used to compute the 3-variate parametric approximating spline was:

Algorithm 2

1. Set the knots sequence in each direction by using (7).
2. Set the B-splines in each direction by using Algorithm 1
3. Calculate (τ_x, τ_y, τ_z), which corresponds to P_{ijk}, by a coordinate transformation.
4. Calculate the value of $s(\tau_x, \tau_y, \tau_z)$ by using (9).

3. Calibration of an articulated robot

3.1 Case studied

A Microrobot 88-5 robot manufactured by Nakanippon Electrics was used for calibration tests. It is a small articulated manipulator with five degrees of freedom (DOF) (see *Figure 1*). The robot was connected with an Olivetti M24 PC and commanded off-line by giving the cartesian coordinates (x,y,z) and the Euler

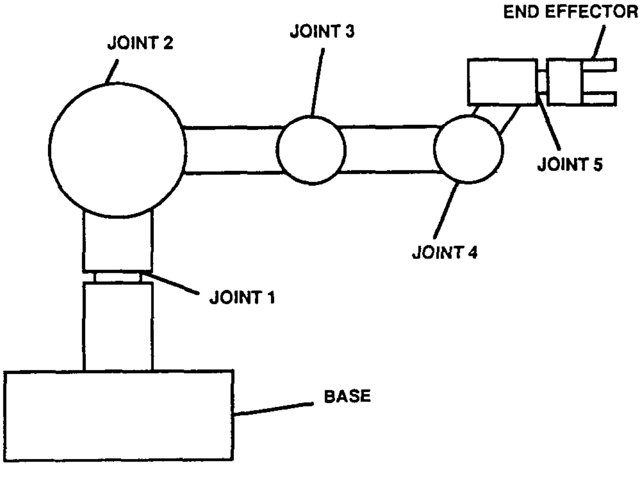

Figure 1. Kinematic scheme of Microrobot 88-5 robot.

angles (ϕ,θ,ψ) of the desired pose. Angle ϕ is always set to arctan (y/x) because the robot has only five DOF. To assess the effectiveness of multivariate spline approximation in the calibration problem we decided to perform the pre-corrective inverse calibration of the robot in a portion of its workspace.

The end-effector pose is affected by position and orientation deviations. Each deviation includes a stochastic component that cannot be reduced by calibration and a systematic component that depends on the five location parameters of the robot. To ease the calibration test we decided to carry out the calibration of the end-effector position for fixed values of Euler angles ($\theta = 90°$, $\psi = 180°$). In this way the systematic component of each deviation depended on the three parameters (x,y,z), and three-variate splines were used.

3.2 Calibration volume and measurement system

The following calibration volume was first chosen:

$$\begin{cases} 350 \le r \le 400\,mm \\ -15° \le \rho \le 7.5° \\ 200 \le z \le 300\,mm \end{cases} \tag{10}$$

The origin and the z-axis of the cylindrical coordinates system coincide with the origin and the z-axis of cartesian coordinates in which the end-effector position is specified. The dimension in the r direction could not be increased because of the dimension of the robot workspace.

In this calibration volume four meridian sections (with 7.5° spacing) were identified and in each (r,z) plane a square mesh (with a side of 50 mm) was established with three points in the z direction and two points in the r direction (see *Figure 2*).

The second calibration test was carried out on a larger calibration volume with $-22.5° \le \rho \le 22.5°$ and 42 knots points. To measure the actual end-effector position before and after the calibration, a measurement apparatus with an accuracy of 0.05 mm was developed. This equipment consists of a fixture with three perpendicular slides in x, y, and z directions that can position a small parallelepiped block in the calibration volume.

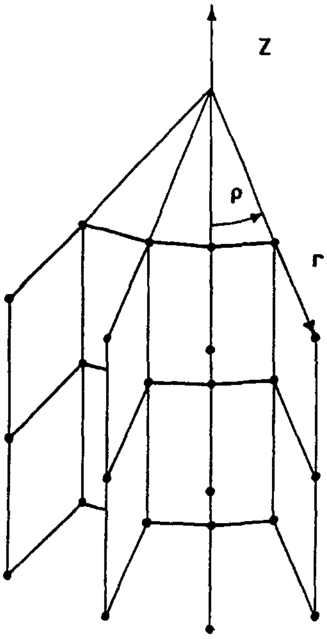

Figure 2. Training points in the calibration volume.

The end-effector of the robot was replaced by a rod with a spherical tip on the tool center point; a weight could be mounted on the rod to simulate the payload. Both the manipulator and the fixture were mounted on a stiff reference plane and each component of the end-effector position was measured moving the slides of the fixture until one of the faces of the small block touched the spherical tip of the robot.

To generate the knots, on which the approximation is based, the robot was located in all the points of the mesh (*training points*). The actual location was measured and the position deviations ($\delta_x,\delta_y,\delta_z$) were calculated. Calibration can reduce systematic deviations depending on robot configuration but cannot affect stochastic deviations. Therefore the best accuracy of a well-calibrated robot coincides with its repeatability. Hence, before making any test, we decided to carry out a repeatability test locating 10 times the end effector in the same position (r = 375 mm, ρ = −3.75°, z = 228 mm). The histograms of the measured deviations with respect to the normal position are reported in *Figures 3, 4* and *5* where the deviations were subdivided in classes with width 0.05 mm and where *cumulative count* means the number of measurements with deviations within a certain class.

The value of repeatability in each direction (in agreement with The National Machine Tool Builder's Association (NMTBA) definition of repeatability in Ref. 14) was taken as 3σ (where σ is the standard deviation), and the following results were obtained:

x direction: repeatability = ±0.14 mm
y direction: repeatability = ±0.12 mm
z direction: repeatability = ±0.26 mm

The offsets of the distributions from 0, caused by systematic deviations, are evident.

3.3 Deviation estimate

This test was aimed to verify the validity of the approximation of the deviations and was carried out with a payload of 0.5 kg on the end-effector of the manipulator. Twelve positions in the calibration volume, located among the knots, were selected and by means of the computer program the approximated values of the position deviations in those points were calculated. The knots in direction ρ, z, and r respectively were 4, 3, and 2; thus the three-variate splines of order 2 2 2, 3 3 2, and 4 3 2 could be used.

In *Tables 1*, *2* and *3* the calculated deviations and the measured deviations are compared for x, y, and z directions respectively. The measured deviation is the difference between the actual position (measured with a maximum error of 0.05 mm) and the position com-

Table 1. Deviations in X direction.

Coordinates			Measured deviation	Calculated deviation		
ρ	z	r		Splines 222	Splines 332	Splines 432
-11.25	228	375	-0.16	-0.054	-0.051	-0.051
-3.75	228	375	-0.20	-0.075	-0.070	-0.053
+3.75	228	375	-0.10	-0.030	-0.029	-0.011
-11.25	278	375	-0.01	+0.055	+0.054	+0.059
-3.75	278	375	-0.05	+0.009	+0.014	+0.034
+3.75	278	375	-0.05	+0.041	+0.044	+0.061
-11.25	203	400	+0.02	+0.030	+0.025	+0.033
-3.75	203	400	+0.06	-0.015	-0.015	+0.011
+3.75	203	400	+0.11	+0.045	+0.038	+0.059
-11.25	303	400	+0.02	+0.180	+0.173	+0.180
-3.75	303	400	+0.06	+0.115	+0.115	+0.137
+3.75	303	400	+0.11	+0.140	+0.137	+0.155

Table 2. Deviations in Y direction.

Coordinates			Measured deviation	Calculated deviation		
ρ	z	r		Splines 222	Splines 332	Splines 432
-11.25	228	375	+0.49	+0.385	+0.372	+0.438
-3.75	228	375	+0.38	+0.328	+0.318	+0.300
+3.75	228	375	-0.08	+0.194	+0.200	+0.114
-11.25	278	375	+0.39	+0.301	+0.291	+0.346
-3.75	278	375	+0.43	+0.275	+0.265	+0.236
+3.75	278	375	+0.02	+0.128	+0.136	+0.048
-11.25	203	400	+0.52	+0.455	+0.449	+0.523
-3.75	203	400	+0.36	+0.405	+0.405	+0.389
+3.75	203	400	-0.06	+0.290	+0.303	+0.212
-11.25	303	400	+0.12	+0.180	+0.179	+0.230
-3.75	303	400	+0.16	+0.170	+0.170	+0.131
+3.75	303	400	-0.26	+0.005	+0.023	-0.071

Table 3. Deviations in Z direction.

Coordinates			Measured deviation	Calculated deviation		
ρ	z	r		Splines 222	Splines 332	Splines 432
-11.25	228	375	-1.77	-1.675	-1.652	-1.680
-3.75	228	375	-1.79	-1.705	-1.673	-1.683
+3.75	228	375	-1.80	-1.765	-1.728	-1.711
-11.25	278	375	-1.79	-1.841	-1.819	-1.834
-3.75	278	375	-1.80	-1.874	-1.842	-1.844
+3.75	278	375	-1.84	-1.905	-1.871	-1.858
-11.25	203	400	-1.70	-1.645	-1.644	-1.681
-3.75	203	400	-1.67	-1.635	-1.635	-1.660
+3.75	203	400	-1.55	-1.725	-1.715	-1.705
-11.25	303	400	-1.37	-1.560	-1.560	-1.578
-3.75	303	400	-1.44	-1.560	-1.560	-1.560
+3.75	303	400	-1.37	-1.560	-1.560	-1.542

manded by means of computer with two decimal digit accuracy.

In the x direction the measured deviations in the knots and in the test points have the same order of magnitude as stochastic deviations (±0.14 mm). Therefore, the spline functions have to model a phenomenon strongly affected by stochastic effects and the results show that the calculated deviations are only a rough estimate of measured deviations.

In the y and z directions the measured deviations are larger than stochastic deviations (±0.12 mm and ±0.26 mm respectively) and the spline functions give a good approximation of the deviations on the test points. The increment of the spline order in ρ and z directions causes, in most cases, small modifications of

Table 4. Approximation errors.

Spline order	2 2 2	3 3 2	4 3 2
y direction	0.17	0.18	0.13
z direction	0.11	0.11	0.11

the estimates. The presence of stochastic deviations in the measurements makes the evaluation of these small modifications difficult.

Nevertheless, to evaluate the variation of the approximation error with the order of the splines, the root-mean-square values of the differences between the estimated deviations and the measured deviations were calculated. The results (see *Table 4*) show that a small improvement in the approximation can be obtained with 4 3 2 splines. If the knot points deviations were measured many times and the approximation were based on the mean value, the mean value would be less affected by stochastic deviations. In this case the spline functions will give a more accurate approximation of the mean value of measured deviations. This approach is interesting in theory but cannot be used in practical applications.

It must be pointed out that stochastic deviations would have caused problems even though a polynomial approximation was used. Moreover, since spline functions give a local approximation, a large stochastic deviation in a knot causes only local errors.

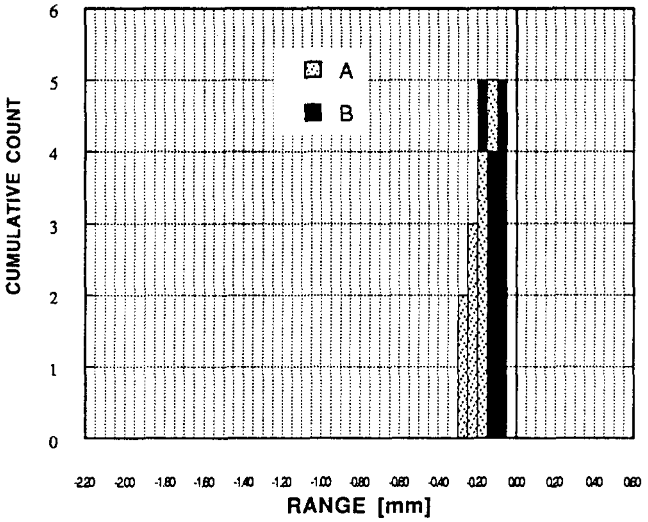

Figure 3. Deviations before calibration (A) and errors after calibration (B) in *x* direction.

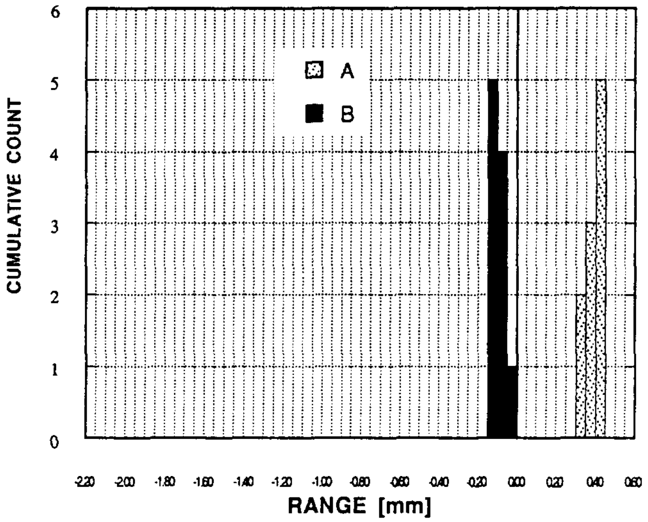

Figure 4. Deviations before calibration (A) and errors after calibration (B) in *y* direction.

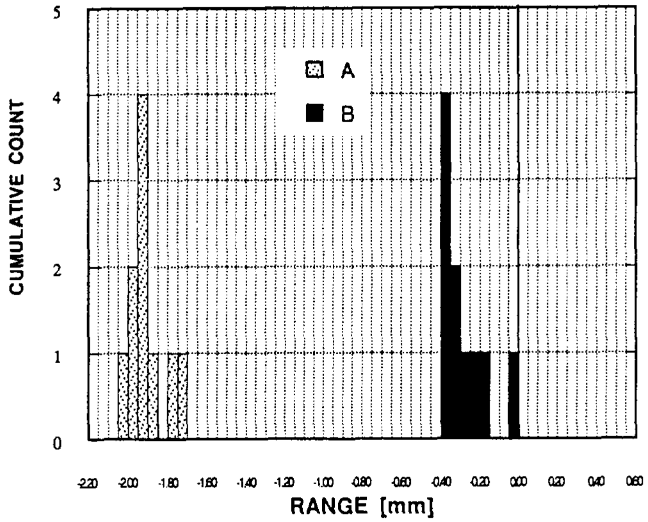

Figure 5. Deviations before calibration (A) and errors after calibration (B) in *z* direction.

Table 6. Test points chosen for calibration without payload.

Points	ρ	z	r
1	−21	278	390
2	−21	228	390
3	−18.75	303	400
4	−18.75	278	375
5	−3.75	278	375
6	3.75	303	400
7	11.25	228	375
8	18.75	228	375

Table 5. Effect of calibration with payload.

	Measured deviation before calibration			Measured error after calibration		
Test point	x	y	z	x	y	z
1	−0.16	+0.49	−1.77	−0.16	+0.19	−0.02
2	−0.20	+0.38	−1.79	−0.07	−0.09	−0.37
3	−0.10	−0.08	−1.80	−0.05	−0.18	+0.15
4	−0.01	+0.39	−1.79	−0.01	+0.09	−0.17
5	−0.05	+0.43	−1.80	0.00	+0.08	−0.20
6	−0.05	+0.02	−1.84	0.00	−0.13	+0.30

Table 7. Effect of calibration without payload.

Test point	Measured deviation before calibration			Measured error after calibration		
	x	*y*	*z*	*x*	*y*	*z*
1	+0.35	+0.91	−0.10	0.00	+0.21	−0.10
2	+0.35	+0.96	−0.10	0.00	+0.16	+0.05
3	+0.23	+0.68	+0.35	−0.02	+0.03	−0.20
4	+0.25	+0.99	0.00	+0.05	+0.19	−0.15
5	−0.20	+0.48	+0.05	0.00	+0.18	−0.10
6	−0.14	−0.31	+0.55	+0.01	−0.06	+0.05
7	−0.01	−0.26	+0.10	−0.01	+0.09	−0.10
8	−0.02	−0.59	+0.20	−0.12	+0.11	−0.30

3.4 Calibration tests

The deviations provided by the approximation algorithm were then used to calibrate robot positioning. The approximation with spline functions of orders 2 2 2 was used. First the calibrated robot, with a payload of 0.5 kg, was commanded to reach six poses (which coincided with the first six poses of the test in section 3.3) and the positioning errors were measured. *Table 5* summarizes the results of calibration, comparing the deviation before calibration with the errors after calibration on the test points chosen.

The calibration was efficacious and the errors in the three directions were reduced toward the range of the repeatability errors. To confirm the effect of calibration the robot was located 10 times in the second test point ($r = 375$ mm, $\rho = -3.75°$, $z = 228$ mm) and the histograms of the errors after calibration in the three directions were drawn. In *Figures 3, 4,* and *5* the histograms are compared with those obtained before calibration. A remarkable reduction of systematic errors due to calibration is evident, even though the variance of the distribution is not modified in a sensible way. The second calibration test was carried out without payload on the end-effector. A larger calibration volume was considered ($-22.5° \leq \rho \leq 22.5°$) and the deviations were measured in 42 training points.

The deviations in z direction were very different from those measured in the first test. The other components were less affected by the absence of the payload, as could be expected. The calibrated robot was commanded to reach eight poses located among the knots (see *Table 6*) and the position errors were measured. *Table 7* shows that in this case also the errors were reduced by calibrations within the range of stochastic errors.

4. Conclusion

The algorithms used to calculate B-splines functions and multivariate parametric approximating splines were applied in an inverse robot calibration problem. Spline functions of order 2 gave a good estimate of position deviations of the end-effector but stochastic deviations prevented a remarkable improvement of the approximation when the order of the spline was increased. Stochastic deviations are related to robot repeatability and higher order splines could be useful for the calibration of robots with better repeatability. Splines of order 2 were then selected for inverse calibration tests. The results showed that robot accuracy was strongly enhanced and the effectiveness of the inverse calibration method was confirmed.

5. Acknowledgement

The authors wish to acknowledge the useful work done by two graduating students in their theses: S. Celli: *Sulla taratura inversa di un robot* and S. Conti: *Taratura diretta e inversa di un manipolatore a cinque gradi di libertà*. The authors are grateful to Italtecnica OMS of Milano (Italy) for providing the robot used to carry out the research.

References

1 Wang, C. B., Chen, J., and Yang, J. C. S. Robot positioning accuracy improvement through kinematic parameter identification. *Third Canadian CAD/CAM and Robotics Conference*. Toronto, Canada, 1984

2 Lozinski, C. A., Whitney, D. E., and Rourke, J. M. Industrial robot forward calibration: Method and result. *ASME Journal of Dynamical Systems, Measurements and Control* 1986, **108**(1), 1–8

3 Roth, Z. S., Mooring, B. W., and Driels, M. R. *Fundamental of Manipulator Calibration*. John Wiley and Sons, New York, 1991

4 Foulley, L. P. and Kelly, R. B. Improving the precision of a robot. *IEEE First International Conference on Robotics*. Atlanta, GA, 1984

5 Shamma, J. S. and Whitney, D. E. A method for inverse robot calibration. *Trans. ASME* 1987, **109**, 36–43

6 Powell, M. J. D. *Approximation Methods and Theory*. Cambridge University Press, Cambridge, UK, 1981

7 Cox, M. G. The numerical evaluation of B-splines. *Journal of IMA* 1972, **10**, 134–149

8 Cox, M. G. The numerical evaluation of a spline from its B-splines representation. *Journal of IMA* 1978, **21**, 135–143

9 De Boor, C. *A Practical Guide to Splines*. Number 27 in Applied Mathematics Sciences. Springer-Verlag, New York, 1978

10 Cox, M. G. An algorithm for spline interpolation. *Journal of IMA* 1975, **15**, 95–108

11 Cox, M. G. A data fitting package for the non-specialist user. *Software for Numerical Mathematics*, Ed. Evans, London, 1974

12 Rogers, D. F. and Adams, J. A. *Mathematical Elements for Computer Graphics*. McGraw-Hill, 1989

13 De Marchi, S. Approssimazione con splines multivariate. Università di Padova, Tesi di Perfezionamento in Matematica Applicata, 1991

14 Klafter, R. D., Chmielewski, T. A., and Negin, M. *Robotic Engineering: An Integrated Approach*. Prentice-Hall International Inc., Englewood Cliffs, NJ, 1989