IMPROVEMENT OF LOCAL POSITION ACCURACY OF ROBOTS FOR OFF-LINE PROGRAMMING

Jin-Hwan Borm* and Jong-Cheon Choi*

(Received September 5, 1991)

For the implementation of industrial robots in a CIM environment, it is necessary to be able to position their end-effectors to an abstractly defined cartesian position with desired accuracy. In other words, it is necessary to find accurate actuator command values corresponding to given goal positions which are expressed with respect to a certain coordinate frame. If the teaching-by-doing method is used, very accurate actuator command values are obtained from transducer readings. For the case when the goal poitions are mathematically expressed, however, the actuator command values for the goal positions must be calculated using robot kinematics. It is, however, well known that the position errors in the order of 10mm is not unusual while many industrial robots have the repeatability in the order of 0.1mm. In here, the position error is referred to as the difference between the specified goal position and the position where a robot is actually controlled. To reduce the position errors, many researchers proposed calibration methods which are based on robot kinematic identification. However, those methods are quite complex and require an accurate position measuring device. In this paper, a new method which does not require the accurate kinematic identification, is introduced. In this method, the accurate actuator command values are calculated using the nominal kinematic model which is appropriatly altered based on the available encoder readings of the several reference frames. To demonstrate the simplicity and the effectiveness of the method, computer simulations as well as experimental studies are performed and their results are discussed.

Key Words: CIM Environment, Robot Position Accuracy Improvement, Local Calibration

1. INTRODUCTION

The ultimate objective of the robot calibration is to find the accurate actuator command values corresponding to the hand positions of a robot which are mathematically defined. Recently, many researchers proposed robot calibration techniques which are based on robot kinematic identification (Borm and Menq, 1991, 1989, 1988 : Menq and Borm, 1989). From the fact that the primary cause of the position errors lies in the discrepancies between the nominal kinematics used in the robot control software and the actual one, this method first tries to identify the more accurate functional relationships (robot kinematics) between the position of the end-effector and the actuator transducer readings. Based on the identified functional relationship which may be referred to as the calibrated robot kinematics, the more accurate actuator command values can then be obtained from the inverse solution of the calibrated kinematics. This procedure may be called global calibration procedure in the sense that it globally improves position accuracy over the robot work space. Even though this procedure is quite general and useful, it is rather complex and costly for some tasks which require only a small portion of the working volume. And it requires an accurate position measuring device and a time-consuming iterative inverse kinematics solver. Moreover the accuracy of this calibration technique is much dependent on the position error modeling (chen and chao, 1986 : Judd and Knasinski, 1987). Since it is impossible to identify or estimate the exact functional relationship, it is inevitable that there will be a residual position error in addition to the repeatability error at any given robot configuration. From the experimental results in may publications (Borm and Menq, 1989 : Judd and Knasinski, 1987 : Veitschegger and Wu, 1987 : whitney et al., 1984), it can be seen that the root mean square (RMS) value of the residual position errors after global calibration is 0.54mm for a robot with 0.31mm maximum repeatability error and 0.2mm−0,5mm for PUMA robot with 0.1mm repeatability error.

In this paper, a different positioning accuracy improvement technique, which uses the nominal inverse kinematic solution and does not require a position measuring machine, is introduced for the case in which goal positions can be expressed with respect to one or more task reference frames. The method to be discussed in this paper is directed at obtaining more accurate actuator commands by directly compensating for the position errors of the goal position by the use of the available encoder readings of the task reference frames. In other words, the task reference frames are first taught to a robot, the corresponding actuator commands are then used to calculate the desired actuator commands for the goal position in conjunction with an interpolation scheme based on a nominal kinematic model. This interpolation scheme will result in acceptable within a desired working area. In fact, this method can also be applied to robots which have already been calibrated using the global calibration technique. In this case, the position accuracy will be further improved in the limited working area. To demonstrate the simplicity and effectiveness of the method, computer simulations as well as experimental studies are performed and their results are discussed.

*Department of Mechanical Engineering, Ajou University, 5 Wonchun Dong Gweonseon-Gu, 440-749, Suwon.

2. LOCAL POSITION ACCURACY IMPROVEMENT BY TEACHING ONE TASK REFERENCE FRAME

When the goal positions of a task are abstractly expressed with respect to the task reference frame, it has been shown (Menq and Borm, 1989) that the error field of the robot can be altered by teaching the task reference frame to the robot. As shown in Fig. 1, the robot reference frame is denoted by [0], the goal frame by [G], and the end-effector position by [G′]. In the case in which the working area of a task is relatively small, as compared to the robot's working volume, one can set up a working or task reference frame, denoted by [W], in which all goal positions are expressed. By teaching the task reference frame [W] to the robot, the joint variables (θ_w) corresponding to the [W] frame can be obtained from the encoder readings. The real transformation from [0] to [W], which is denoted by $T^r(\theta_w)$, be slightly different from a tranformation $T^c(\theta_w)$ which is calculated by using the known kinematic model of the robot. This difference can be expressed as.

$$T^r(\theta_w) - T^c(\theta_w) = T^c(\theta_w)\cdot{}^{w}\Delta \tag{1}$$

Where, ${}^{w}\Delta$ is the differential transformation matrix or error matrix in terms of the coordinate frame [W] and can be expressed as:

$${}^{w}\Delta = \begin{bmatrix} 0 & -\delta_z^w & \delta_y^w & d_x^w \\ \delta_z^W & 0 & -\delta_x^w & d_y^w \\ -\delta_y^w & \delta_x^w & 0 & d_z^w \\ 0 & 0 & 0 & 0 \end{bmatrix} \tag{2}$$

where d_x^w, d_y^w, d_z^w, δ_x^w, δ_y^w and δ_z^w are the differential translations and rotations in terms of the coordinate frame [W].

In the off-line programming application, the transformation from the working reference frame [W] to the goal position [G] is known and denoted by ${}^{w}T_G$. Consequently, the real transformation from [0] to [G] becomes $T^r(\theta_w)\cdot{}^{w}T_G$, which represents the goal position to which one wants to control the end-effector. The control of displacement of the robot to the goal position requires the joint variables(θ_G),

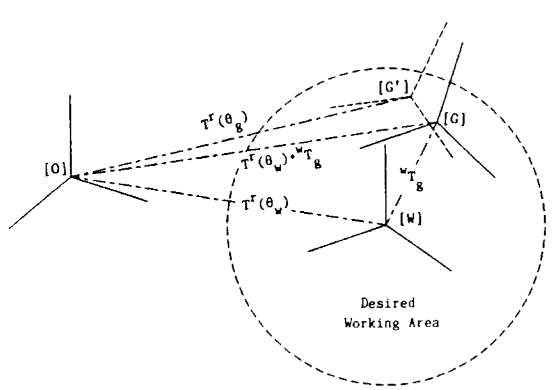

Fig. 1 Task working area and related coordinate frames

which should be calculated from the inverse kinematic solution of the following equation,

$$T^r(\theta_G) = T^r(\theta_W)\cdot{}^{w}T_G \tag{3}$$

However, $T^r(\theta_w)$ is not known while $T^c(\theta_w)$ is the known kinematic model of the robot. Substituting Eq. (1) to Eq. (3), one has,

$$T^c(\theta_G)\cdot[I+{}^{G}\Delta] = T^c(\theta_W)\cdot[I+{}^{W}\Delta]\cdot{}^{W}T_G \tag{4}$$

Eq. (4) can then be expressed as,

$$T^c(\theta_G) = T^c(\theta_W)\cdot{}^{W}T_G + [T^c(\theta_W)\cdot{}^{W}\Delta\cdot{}^{W}T_G - T^c(\theta_G)\cdot{}^{G}\Delta] \tag{5}$$

where $T^c(\theta_G)\cdot{}^{G}\Delta$ is the cartesian error of the end effector at the coordinate frame [G] which is seen in frame [G] and $T^c(\theta_W)\cdot{}^{W}\Delta\cdot{}^{W}T_G$ is the cartesian error of the end effector at frame [W] which is seen in the frame [G]. Since the second term in the Eq. (5), $[T^c(\theta_W)\cdot{}^{W}\Delta\cdot{}^{W}T_G - T^c(\theta_G)\cdot{}^{G}\Delta]$, is the difference between the errors at [G] and [W], it is the second order term of the error. If the coordinate frame [G] and [W] are closely located, the difference between the error in [G] and that in [W] becomes quite small. From the fact that the second term in Eq. (5) is small, the following equation is suggested to obtain the approximate actuator commands (θ_G) corresponding to the goal position, ${}^{W}T_G$.

$$T^c(\theta_G) = T^c(\theta_W)\cdot{}^{W}T_G \tag{6}$$

When goal position [G] is located near to the origin of [W], a quite accurate inverse solution for the real robot can be obtained using Eq. (6). If the goal position is the taught reference frame itself, then ${}^{W}T_G$ becomes the identity matrix [I], and from Eq. (6), (θ_G) becomes (θ_W) which is the exact inverse solution for the given goal position [W].

3. LOCAL POSITION ACCURACY IMPROVEMENT BY TEACHING K TASK REFERENCE FRAMES

In most of off-line programming applications, the goal positions can be expressed with respect to many task reference frames. If this is the case and the task reference frames are taught to a robot, the position accuracy of the goal positions can be improved using the taught information. During teaching a robot K task reference frames $[W_1]$, $[W_2]$, ..., $[W_k]$, the joint variable vectors $\theta_{w1}, \theta_{w2}, \ldots, \theta_{wk}$ corresponding to the taught frames can be obtained from the encoder readings. The differences between the real transformation and the calculated one from [0] to the taught reference frames can be expressed as,

$$\begin{aligned} T^r(\theta_{w1}) - T^c(\theta_{w1}) &= T^c(\theta_{w1})\cdot{}^{w1}\Delta \\ T^r(\theta_{w2}) - T^c(\theta_{w2}) &= T_c(\theta_{w2})\cdot{}^{w2}\Delta \\ &\vdots \\ T^r(\theta_{wk}) - T^c(\theta_{wk}) &= T(\theta_{wk})\cdot{}^{wk}\Delta \end{aligned} \tag{7}$$

where ${}^{wi}\Delta$ is the differential transformation matrix in terms of the coordinate frame $[W_i]$. In off-line programming applications, the transformations from the taught task reference frames $[W_1, W_2, \ldots, W_k]$ to a goal position $[G]$ are known and denoted by ${}^{w1}T_G$, ${}^{w2}T_G$, and .... ${}^{wk}T_G$. The real transformation from [0] to [G] can then be expressed amny different forms depending on which reference frame is used to expressed the goal position.

$$\begin{aligned} T_G^r &= T^r(\theta_{w1}) \cdot {}^{w1}T_G \\ &= T^r(\theta_{w2}) \cdot {}^{w2}T_G \\ &\vdots \\ &= T^r(\theta_{wk}) \cdot {}^{wk}T^G \end{aligned} \tag{8}$$

These expressions are intended to represent the same actual goal position with respect to a robot reference frame to which one likes to control the robot. The control of the robot to this goal position requires the joint variables(θ_G). If the real transformation $T^r(\theta)$ were known, the exact actuator command variables(θ_G) could be found using one of the following equations,

$$\begin{aligned} T^r(\theta_G) &= T^r(\theta_{w1}) \cdot {}^{w1}T_G \\ T^r(\theta_G) &= T^r(\theta_{w2}) \cdot {}^{w2}T_G \\ &\vdots \\ T^r(\theta_G) &= T^r(\theta_{wk}) \cdot {}^{wk}T_G \end{aligned} \tag{9}$$

Since the real transformation, $T^r(\theta)$ is not known, the inverse solution of the above equation cannot be obtained. If the task frames can be taught to the robot, the joint variables (θ_G) can be calculated by incorporating the information from the encoder readings of the taught frames via selected weighting coefficients such that the resulting position errors become zero at the taught positions, and becomes smaller at the other goal positions. If the weighting values $u_1, u_2, \ldots, u_k$ are multiplied into both sides of Eq. (9), and all the equations are added, then,

$$\begin{aligned} &T^r(\theta_G) \cdot [u_1 + u_2 + \cdots + u_k] \\ &= T^r(\theta_{w1}) \cdot {}^{w1}T_g \cdot u_1 + T^r(\theta_{w2}) \cdot {}^{w2}T_G \cdot u_2 + \cdots \\ &\quad + T^r(\theta_{wk}) \cdot {}^{wk}T_G \cdot u_k \end{aligned} \tag{10}$$

If the position errors are relatively small, then,

$$T^r(\theta_G) = T^c(\theta_G) + T^c(\theta_G) \cdot {}^G\Delta \tag{11}$$

Substituting Eqs (7) and (11) to Eq. (10), then

$$\begin{aligned} &T^c(\theta_G) \cdot [I + {}^G\Delta][u_1 + u_2 + \cdots + u_k] \\ &= T^c(\theta_{w1}) \cdot [I + {}^{w1}\Delta] \cdot {}^{w1}T_G \cdot u_1 + \\ &\quad T^c(\theta_{w2}) \cdot [I + {}^{w2}\Delta] \cdot {}^{w2}T_G \cdot u_2 + \\ &\quad \cdots + T^c(\theta_{wk}) \cdot [I + {}^{wk}\Delta] \cdot {}^{wk}T_G \cdot u_k \end{aligned} \tag{12}$$

In the same manner as in the previous section, from Eq. (12), the following equation can be used to botain the approximate actuator commands (θ_G) corresponding to the goal position for the case in which K task reference frames are used.

$$\begin{aligned} &T^c(\theta_G) \cdot [u_1 + u_2 + \cdots + u_k] \\ &= T^c(\theta_{w1}) \cdot {}^{w1}T_G \cdot u_1 + \\ &\quad T^c(\theta_{w2}) \cdot {}^{w2}T_B \cdot u_2 + \\ &\quad \cdots + T^c(\theta_{wk}) \cdot {}^{wk}T_G \cdot u_k \end{aligned} \tag{13}$$

where $T^c(\theta)$ is he known kinematic transformation with which the inverse kinematic solution can be easlily obtained. When a task working volume is relatively small and the task reference frames are chosen close to each other, Eq. (13) will then provide quite accurate inverse kinematic solution for any goal positions encircled by the taught task reference frames.

However, the accuracy of the inverse of Eq. (13) will be dependent on the values of the weighting coefficients $(u_1, u_2, \ldots, u_k)$. In fact, in Eq. (13), the position errors of the goal position are compensated by the combination of the weighted taught information. If a goal position is very near to a taught reference frame, say $[W_i]$, then the corresponding weighting value $[u_i]$ must be much higher than any other weighting values. Therefore, possible variables for the weighting function could be the distances from the taught points to the goal position. In this paper, the following form of the weighting function $[u_i]$ is suggested :

$$u_i(d_1, d_2, \cdots\cdots, d_k) = \frac{C_0 + C_{i1}d_i^2 + \cdots + C_{ik}d_k^2}{d_1^2 + \cdots + d_k^2} \tag{14}$$

where $C_0, C_{i1}, C_{i2}, \cdots C_{ik}$ are constant values to be calculated by assigning some boundary conditions, and d_i is the distance from the orgin of $[W_i]$ to the goal position $[G]$. When a goal position $[G]$ is one of the taught reference frames itself, say $[W_i]$, then di becomes zero. In this case, u_i must be 1 while all the other weighting values, $u_j (j \neq i)$ are zero. Therefore, the suggested weighting functions (u_i) must satisfy the following boundary conditions.

(a) For $i = 1, 2, \ldots, K$,

$$u_i(D_{j1}, D_{j2}, \cdots, D_{jk}) = \begin{bmatrix} 1 \text{ when } D_{ji} = 0 \\ 0 \text{ when } D_{ji} = 0 \end{bmatrix} \quad j = 1, 2, \cdots, \; k \tag{15}$$

and,

$$\text{(b)} \quad \sum_{i=1}^{k} u_i = 1 \tag{16}$$

where D_{ij} is the distance from the origin of the frame $[W_i]$ to the origin of $[W_j]$, and $D_{ij} = D_{ji}$ and $D_{ji} = 0$ when $i = j$. When the number of taught frames to be used is K, the number of the weighting functions is K and each weighting function has K+1 unknown costants as shown in Eq. (14). However, in total, the number of unknowns is (K^2+1), and those are C_0 and $C_{i1}, C_{i2}, \ldots, C_{ik} (i = 1, 2, \ldots, K)$. The number of the boundary conditions is also (K^2+1). By applying the $B.C$'s of Eq. (15) for $i = 1, 2, \ldots K$ to Eq. (14), we have K^2 equations and one more from Eq. (16). Therefore the unknown costants can be solved.

Now, for example, some of the weighting functions are evaluated. When K is 1, it is obvious that u_1 is 1. When K is 2, u_1 and u_2 can be obtained from Eqs. (14), (15) and (16), and found to be :

$$u_1(d_1, d_2) = \frac{d_1^2}{d_1^2 + d_2^2} \tag{17}$$

$$u_2(d_1, d_2) = 1 - u_1(d_1, d_2) \tag{18}$$

For the case in which K is 3, the three weighting functions u_1, u_2, and u_3 are obtained as :

$$\begin{aligned} u_1(d_1, d_2, d_3) = \frac{(D_{12}^2 + D_{13}^2)}{\text{A}} \Big[&\frac{-D_{23}^2}{D_{12}^2 D_{13}^2} d_1^2 \\ &+ \frac{1}{D_{12}^2} d_2^2 + \frac{1}{D_{13}^2} d_3^2 \Big] \end{aligned} \tag{19}$$

$$u_2(d_1,d_2,d_3)=\frac{(D_{12}^2+D_{23}^2)}{A}\left[\frac{1}{D_{12}^2}d_1^2+\frac{-D_{13}^2}{D_{12}^2D_{23}^2}d_2^2+\frac{1}{D_{13}^2}d_3^2\right] \quad (20)$$

$$u_3(d_1,d_2,d_3)=1-u_1(d_1,d_2,d_3)-u_2(d_1,d_2,d_3) \quad (21)$$

where $A=2(d_1^2+d_2^2+d_3^2)$

4. EXPERIMENTS AND SIMULATIONS

Using an RM501 robot with five degrees of freedom, experiments and computer simulations are performed to verify that the use of Eq. (6) or (13) can provide more accurate inverse kinematic solution for a real robot when goal positions are expressed with respect to one or a number of task reference frames and the task reference frames are taught to the robot. For the purpose of simplicity, joints 4 and 5 were locked (fixed) in this experiment, and the robot coordinate system and normal values of its geometric parameters are shown in Fig. 2 and Table 1 respectively.

51 equally spaced end-effector positions and their corresponding actuator commands were measured using 3-D Coordinate Measuring System. The coordinate measuring system

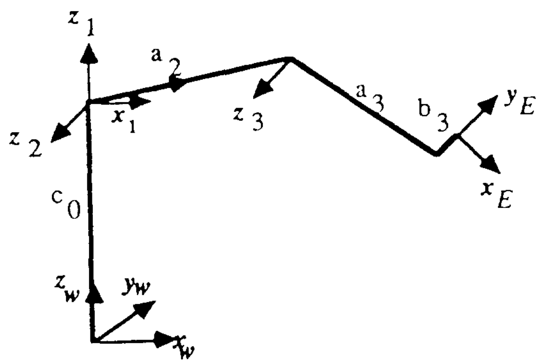

Fig. 2 Coordinate system of RM501 robot with 4 and 5 axes fixed

Table 1 Nominal geometric parameter values for RM501 robot

(Units are mm and degree)

	a	b	c	α	β	γ
W-1	0	0	250	0	0	0
1-2	0	0	0	90	0	0
2-3	220	0	0	0	0	0
3-E	160	65	0	0	0	0

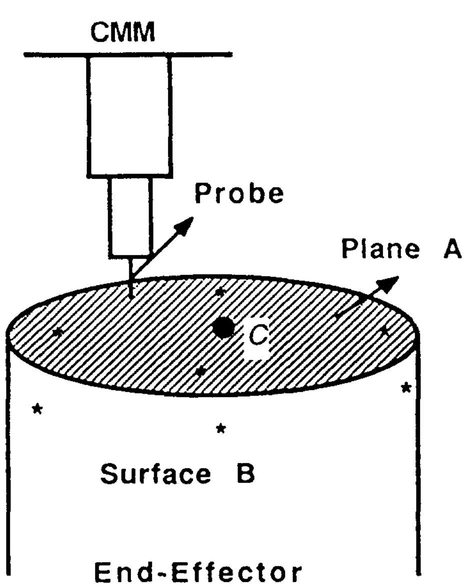

Fig. 3 End-effector of cylindrical shape

Table 2 Inverse kinematic solution before compensation

(in degrees)

Position No.	Exact Experimental Solution			Nominal Inverse Solution		
	θ_1	θ_2	θ_3	θ_1	θ_2	θ_3
15	−10.45	38.725	−83.90	−11.059	43.431	−84.720
16	−10.00	34.45	−74.95	−10.641	39.243	−75.994
17	−9.575	29.425	−64.30	−10.237	34.327	−65.649
18	−9.20	22.80	49.975	−9.856	28.002	−52.040
19	−5.25	39.825	86.20	−5.839	44.525	−87.031
20	−5.025	35.65	77.475	−5.651	40.451	−78.488
21	−4.825	30.80	67.225	−5.465	35.689	−68.462
22	−4.625	24.675	54.050	−5.279	29.773	−55,842
24	0.0	40.20	86.95	−0.578	44.919	−87.767
25	0.0	36.05	78.30	−0.582	40.817	−79.325
26	0.0	31.25	68.175	−0.607	36.118	−69.436
27	0.0	25.25	−55.3	−0.623	30.34	−57.095
Max. error (deg.)				0.662	−5.202	2.065

Table 3 Compensated inverse kinematic solution by utilizing one teaching information at position No.26

(in degrees)

Position No.	Exact Experimental Solution			Compensated Inverse Kinematic. By teaching Position No. 26		
	θ_1	θ_2	θ_3	θ_1	θ_2	θ_3
15	−10.45	38.725	−83.90	−10.168	38.681	−84.386
16	−10.00	34.45	−74.95	−9.797	34.594	−75.459
17	−9.575	29.425	−64.30	−9.435	29.720	−64.812
18	−9.20	22.80	−49.975	−9.093	23.286	−50.590
19	−5.25	39.825	−86.20	−5.054	39.622	−86.501
20	−5.025	35.65	−77.475	−4.905	35.657	−77.760
21	−4.825	30.80	−67.225	−4.754	30.945	−67.432
22	−4.625	24.675	−54.050	−4,600	24.937	−54.224
24	0.0	40.20	−86.95	0.084	39.911	−87.064
25	0.0	36,05	−78.30	0.052	35.907	−78.390
26	0.0	31.25	−68.175	0.00	31.25	−68.175
27	0.0	25.25	−55.3	−0.040	25.367	−55.213
Max. error (deg.)				−0.282	−0.486	0.615

Table 4 Compensated inverse kinematic solution by utilizing two teaching information at position No.17 and 26

(in degrees)

Position No.	Exact Experimental Solution			Compensated Inverse Kinematic. By teaching Position No.17 and 26		
	θ_1	θ_2	θ_3	θ_1	θ_2	θ_3
15	−10.45	38.725	−83.90	−10.312	38.456	−84.038
16	−10.00	34.45	−74.95	−9.941	34.339	−75.044
17	−9.574	29.425	−64.30	−9.575	29.425	−64.30
18	−9.20	22.80	−49.975	−9.228	22.886	−49.834
19	−5.25	39.825	−86.20	−5.134	39.521	−86.352
20	−5.025	35.65	−77.475	−4.982	35.547	−77.588
21	−4.825	30.80	−67.225	−4.830	30.816	−67.218
22	−4.625	24.675	−54.050	−4.674	24.770	−53.923
24	0.0	40.20	−86.95	0.077	39.904	−87.054
25	0.0	36.05	−78.30	0.051	35.906	−78.389
26	0.0	31.25	−68.175	0.00	31.25	−68.175
27	0.0	25.25	−55.3	−0.041	25.366	−55.212
Max. error (deg.)				−0.138	0.304	0.152

Table 5 Compensated inverse kinematic solution by utilizing three teaching information at position No.15, 17 and 26

(in degrees)

Position No.	Exact Experimental Solution			Compensated Inverse Kinematic. By teaching Position No.15,17 and 26		
	θ_1	θ_2	θ_3	θ_1	θ_2	θ_3
15	−10.45	38.725	−83.90	−10.45	83.725	−83.90
16	−10.00	34.45	−74.95	−10.01	34.480	−74.972
17	−9.575	29.425	−64.30	−9.575	29.425	−64.30
18	−9.20	22.80	−49.975	−9.223	22.897	−49.856
19	−5.25	39.825	−86.20	−5,276	39.692	−86.205
20	−5.025	35.65	−77.475	−4.996	35.677	−77.618
21	−4.825	30.80	−67.225	−4.823	30.827	−67.235
22	−4.625	24.675	−54.050	−4.687	24.738	−53.865
24	0.0	40.20	−86.95	0.009	39.954	−86.969
25	0.0	36.05	−78.30	0.024	35.917	−78.358
26	0.0	31.25	−68.175	0.00	31.25	−68.175
27	0.0	25.25	−55.3	−0.048	25.351	−55.186
Max. error (deg.)				0.062	0.133	0.185

consists of a Sheffield RS-30 Cordax Coordinate Measuring Machine (CMM) equipped with an MP-30/35 processor and HP9000 series 300 computer. Linear accuracy of the CMM is specified to be (3+3L) micro-meters which is sufficiently accurate for this experiment. When the position of the end-effector is being directly measured, however, it is qutie difficult to precisely locate the probe tip of the CMM to the tip or center of the end-effector. For this reason, the end-effector of cylindrical shape shown in Fig. 3 was used, and four points on the end plane A and four more points on the cylinder surface are measured. The center point position (C) of the end-effector is then calculated.

Using the experimental data, simulations were performed. The first simulation was to find the nominal inverse kinematic solution for the 51 hand positions without compensation. Some of the results are compared with the experimentally obtained joint angles as shown in Table 2. The errors are quite big especially in θ_2. In Table 2, maximum error in θ_1 is around 0.65 degree, 5.2 (deg) error in θ_2 and 1.8 (deg) error in θ_3. The second simulation was for the case in which the position number 26 is taught to the robot and all other positions are referred to the coordinate frame with origin at position number 26. Using Eq. (6), the inverse solutions of the nominal kinematic model are calculated after the 51 hand positions are compensated. In Table 3, some of the calculated inverse kinematic solutions for the hand positions located near to the taught position are compared with the experimentally obtained inverse solutions. Maximum error in θ_1 is around 0.282 degree, 0.49 (deg) error in θ_2 and 0.615 (deg) error in θ_3. For the positions located near to the taught position, for example position no. 27, error in θ_1 is 0.04 degree, −0.117 (deg) error in θ_2 and 0.87 (deg) error in θ_3. In fact,

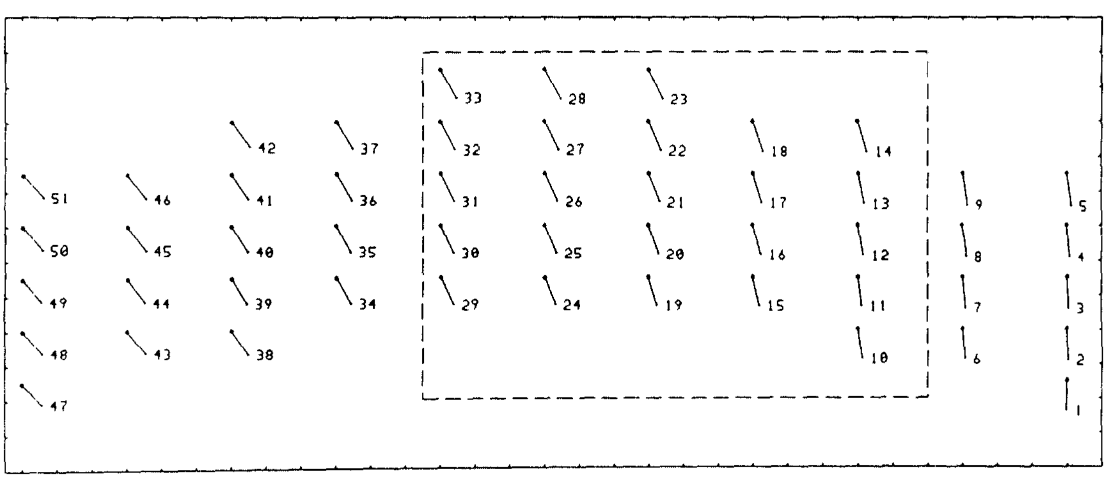

Fig. 4 Uncompensated postion error

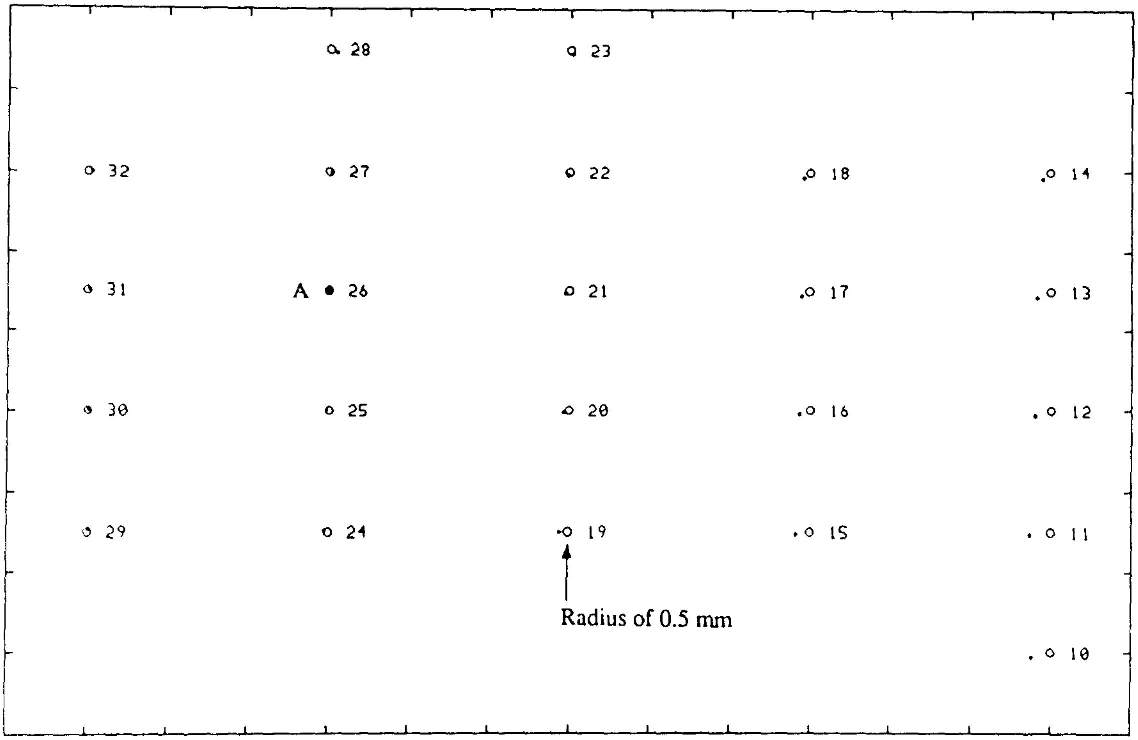

Fig. 5 Position error compensated with 1 taught position (26)

these errors in joint angles cause less than 0.5mm error of the end-effector at that position, which is comparable to the maximum value of the robot repeatability errors. It is clearly verified that the proposed method can significantly improve the accuracy of the inverse kinematic solution. Table 4 shows some of the results of nominal inverse kinematic solutions for the case in which 2 positions (17, 26) are taught to the robot and the goal positions are compensated with respect to the information from those positions. The inverse kinematic solutions are calculated by use of Eq. (14) together with the

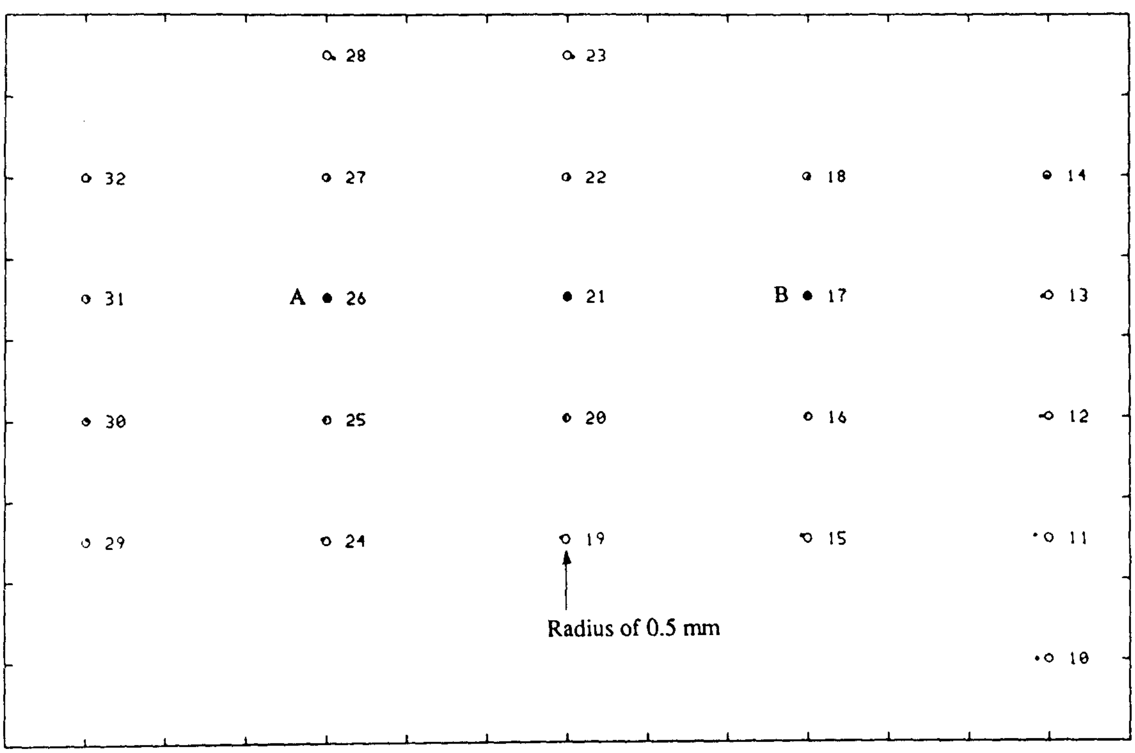

Fig. 6 Position error compensated with 2 taught positions (17, 26)

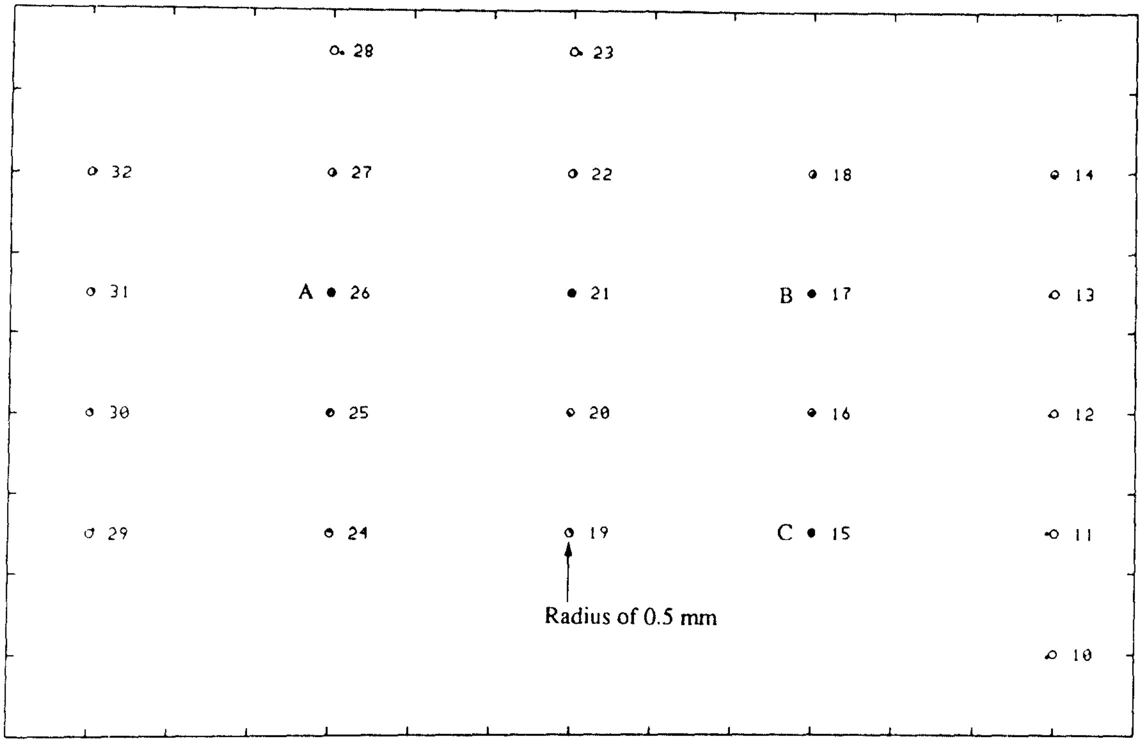

Fig. 7 Position error compensated with 3 taught positions (15, 17, 26)

taught information. It can be seen that the accuracy of the calculated actuator commands is generally improved over that of the case in which only one position teaching information was utilized. Table 5 shows the inverse kinematic solutions for the case in which 3 positions (15, 17, 26) are thaught to the robot. Accuracy in the calculated joint angles are further improved over those of the previous cases.

Position errors due to the errors in the calculated actuator commands can be also simulated. If the errors in the joint angles are small, the position errors can be assumed to be.

$$dT = T^c(\theta_r) - T^c(\theta_c)$$

where the elements of θ_r are the experimentally obtained joint encoder readings, the elements of θ_c are the calculated actuator command values, and $T_c(\theta)$ is the nominal kinematic model. Fig. 4 shows the uncompensated position errors. Figs. 5, 6 and 7 show the compensated position errors for the cases in which the inverse solutions are obtained with one, two and three taught position information respectively. As shown in the figures, the position accuracy is improved as the number of taught positions used is increased.

In fact, the actual position errors occured when the robot is controlled using the calculated inverse solution, will be the simulated position errors plus the repeatability error of the robot. The small circles in Figs. 5, 6 and 7 have a radius of 0.5mm which has been specified as the maximum value of repeatability errors. When the calculated position error is within the small circle, most likely the robot will actually move to the specified position within 0.5mm error.

5. CONCLUSION

A new procedure for locally improving the position accuracy of a robot has been presented in this paper. The method directly compensates for the position errors of a goal position in the limited working area by use of taught information. Since this scheme does not require accurate robot kinematics or large number of measurements, it is much simpler than the conventional technique. Moreover, it has been shown experimentally that the position accuracy obtained using this scheme is quite acceptable and comparable to the robot repeatability error in the limited working area. If calibrated robot kinematics is available, the calibrated inverse kinematics technique presented by Borm (1988), and Vitchegger and *Wu* (1987) can be used to find the actuator command values after the hand position is modified by the proposed method. In this case, the position accuracy will be further improved over a limited working area. The main advantage of this scheme is that the procedure is very simple and can be incorporated with the teaching-by doing method even when the goal positions are mathematically expressed. The disadvantage of the method is that the position accuracy is obtained only in the limited working area in which the teaching is performed. If a task can be separated into small subtasks which require only small working volumes, this local calibration scheme becomes quite attractive. The technique is also important in applications that may involve a rather large number of task points in which reteaching all the task points whenever small changes exist in the robot system is not preferable, as well as in the implementation of a robot in a CIM system.

ACKNOWLEDGEMENT

This research was supported by the office of Research of Ajou University under seed grant program.

REFERENCE

Borm, J.H. and Menq, C.H., 1991, "Determination of Optimal Measurement Configurations for Robot Calibration Based on Observability Measure," International Journal of Robotics Research., Vol. 10, No. 1, pp. 51~63.

Borm, J.H. and Menq, C.H., 1989, "Experimental Study of Observability of Parameter Errors in Robot Calibration," IEEE International Conference on Robotics and Automation, Vol. 1, pp. 587~592.

Borm, J.H., 1988, "Improvement and Characterization of Robot Positioning Accuracy for Off-Line Programming," Ph. D.Dissertation, the Ohio State University.

Chen,J.and Chao,L., 1986, "Positional Error Analysis for Robot Manipulators with all Rotary Joints," IEEE International Conference on Robotics and Automation, San Francisco, pp.1011~1016

Judd,R.P.and Kansinski,A.B., 1987, "A Technique to Calibrate Industrial Robot with Experimental Verification," Prodeedings of IEEE International Conference on Robotics and Automation, Vol. 3, pp 351~357.

Menq, C.H. and Borm, J.H., 1989, "Statistical Characterization of Position Errors of an Ensemble of Robots and its Applications," Trasactions of the ASME Journal of Mechanisms, Transmissions, and Automation in Design, Vol. 3, pp. 215~222.

Menq, C.H., Borm, J.H. and Lai, J.Z., "Indentification and Observability Measure of a Basis Set of Error Parameters in Robot Calibration," Accepted for publication in the ASME Journal of Mechanisms, Transmissions, and Automation in Design.

Paw,R., 1982, "Robot Manipulators : Mathematics, Programming and Control," M.I.T. Press

Veitschegger,W.K. and Wu,C.H., 1987, "A Method for Calibrating and Compensating Robot Kinematic Errors," IEEE International Conference on Robotic and Automation, Vol.1, pp 39~44.

Whitney,D.E., Lozinski,C.A. and Rourke,J.M., 1984, "Industrial Robot Calibration Method and Results," Proceeding for the ASME Conference on Computer and Engineering, pp. 92~100