An Analysis of Inverse Kinematics and Singular Configuration for Six Axes Robot with Wrist Offset (ICEIC'04)

Lee YoungDae* and Cho KumBae**
* Dept of Information and Communication, Semyung University, Jechon City, 390-711, Korea
Tel : +82-043-649-1416 Fax : +82-043-649- 1278 E-mail: youngdae77@yahoo.co.kr
**School of Electronic Eng., Seoul National University, Seoul, 980-1234, Korea

Abstract: The inverse kinematics problem is to find a set of joint variable values that will place the end effector of a robot manipulator into a given pose. Pieper has shown that a sufficient condition for a manipulator to have a closed form solution is that three adjacent joint axes intersects, hence the six axes robot with spherical wrist allows closed form solution. But many industrial robots have a non-spherical wrist to provide a stronger wrist configuration so that they can handle heavy payloads. Also, the use of a non-spherical wrist can result in a cheap and simple wrist arrangement than when all three axes intersect at a common point. In these cases, closed form solutions cannot be found. Therefore numerical technique must be used to solve the inverse kinematics equations. This paper proposes a new algorithm that can be used for finding inverse kinematics solution of the six axes robot with non-spherical wrist. Computer simulations are provided to prove the usefulness of our method.

1. INTRODUCTION

We found out the closed form of inverse kinematics solution by assuming that the PUMA robot has a wrist offset of the 5-th axes or the 6-th axis. Then, we compared it with the conventional Newton-Raphson based iterative search methods. From this investigation, we found some shortcomings of the conventional methods. The conventional method has two shortcomings. They often fall into non-convergent cycle when it saddles on local maximum or minimum and it cannot guarantee to find the solution. Moreover it requires lots of iteration so as to avoid this local optimal point.

Thus, the conventional numerical method does not consider the feasible region of solution, thus it often falls into an unfeasible point. Therefore, the system can be broken down during iterative computational procedure.

So as to solve this problem, we propose a unidirectional search algorithm considering the feasible range of inverse kinematics solution. Our algorithm can not only overcome these shortcomings of the conventional one using the suggested reflection methods on the boundary and curved points but also guarantee to find the solution.

Furthermore, Our algorithm not only can find out the exact solution always without errors but also converges very rapidly within ten iterations comparing with conventional methods.

This paper is organized as follows. In section 2, inverse kinematics problem for the robot with wrist offset is introduced. Section 3 analyzes the kinematics properties of PUMA robot with wrist offset. Section 4 reviews two conventional algorithms and their limitations. In section 4, we suggest novel inverse kinematics algorithm using reflection method. Section 5 shows some illustrations of simulation results comparing our method with two conventional ones. Finally, we conclude our remarks.

2. INVERSE KINEMATICS

2.1. Background

Fig.2.1 shows the link coordinate and its variables using the generalized D-H notation, which is commonly

used in robotics[1].

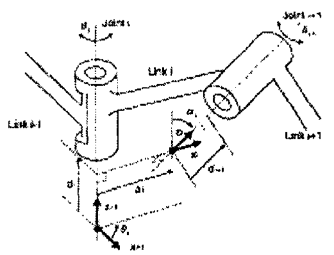


Fig. 2.1 Link coordinate and its variables

where, a_i is the link length, α_i is the link twist angle, d_i is the link offset angle and θ_i is the joint angle, respectively. For convenience, hereafter we will denote
$Cos(\vartheta_i) = C\vartheta_i = C_i$, $Cos(\vartheta_i + \vartheta_j) = C_{ij}$
$Sin(\vartheta_i) = S\vartheta_i = S_i$, $Sin(\vartheta_i + \vartheta_j) = S_{ij}$ and etc.
Then, the homogeneous transformation matrix from the link frame {i} to {i+1} is expressed by

$$ {}^{i-1}_{i}A = \begin{bmatrix} C\theta_i & -C\alpha_i S\theta_i & S\alpha_i S\theta_i & a_i C\theta_i \\ S\theta_i & C\alpha_i C\theta_i & -S\alpha_i C\theta_i & a_i S\theta_i \\ 0 & S\alpha_i & C\alpha_i & d_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{2.1} $$

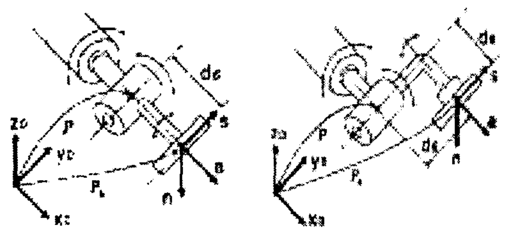
The transformation matrix on forward kinematics from frame {0} to last link frame {6} becomes

$$ {}^{T}_{0}A={}^{0}_{1}A\cdot{}^{1}_{2}A\cdots{}^{5}_{6}A=\begin{bmatrix} n & s & a & p \\ 0 & 0 & 0 & 1 \end{bmatrix}=\begin{bmatrix} n_x & s_x & a_x & p_x \\ n_y & s_y & a_y & p_y \\ n_z & s_z & a_z & p_z \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2.2) $$

where, the vectors n, s, a are related to the robot orientation and p is to the robot position. The inverse kinematics problem is to find the set of joint variables $\theta = \{\theta_1, \theta_2 \cdots \theta_6\}$ from the given transformation matrix ${}^{T}_{0}A$ so as to place the end effector of a robot manipulator into a given pose, while forward kinematics is vice versa.

2.2. Spherical and Non-spherical wrist

Robot wrist can have one, two or three degree of freedom, which can be varied according to its application. Briefly speaking, a spherical wrist without offset corresponds to a special kind of non-spherical wrist without offset The Fig.2.2 shows spherical wrist and non-spherical one with offset, respectively.



(a) (b)
Fig. 2.2 (a) Spherical wrist (b) Non-spherical wrist

2.3. Wrist Offset and Inverse Kinematics

It is well known that a sufficient condition for a manipulator to have a closed form solution is that three adjacent joint axes intersect each other on one point. Accordingly, if six axes robots have a spherical wrist, then they have the closed form of inverse kinematics solution, which makes it very easy to robot position control. The many other researchers have found inverse kinematics solution in this case.

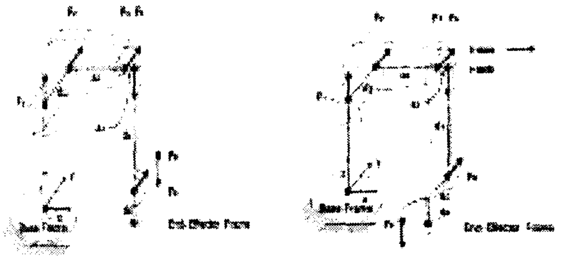
The merit of spherical wrist is in simple representation of kinematics by decoupling position and orientation of a robot manipulator. However, the spherical wrist without offset is not easy to be built and it degrades the load capacity. The wrists with offsets, namely non-spherical wrists, are very common in industrial robots because they are easily manufactured and can handle heavy loads. For this reason, many industrial robots adopt non-spherical wrists, namely wrists with offsets, thus they do not have the closed form of the inverse kinematics because they do not satisfy Pieper's sufficient condition.

The robots with wrist offsets have no closed form of inverse kinematics. It is well known that inverse kinematics without closed form should be solved using the numerical analysis tools such as the multi-dimensional Newton Raphson method.

3. PUMA ROBOT WITH WRIST OFFSET

3.1. PUMA Robot

The commercial PUMA robot of elbow type shown in Fig. 3.1(a) has a spherical wrist and its link parameters can be summarized as the table.1. As mentioned previously, non-spherical wrists are more common in industry and spherical wrist is a special case. In this section, we will assume that PUMA robot has offset like as in Fig.3.1(b) and investigate its properties of inverse kinematics. In the literature [2], several geometric configuration indexes were introduced to solve heuristically its inverse kinematics of PUMA robot with a spherical wrist.



(a) spherical wrist (b) non-spherical wrist
Fig.3.1 PUMA robot of elbow type

Note that the transformation matrix of non-spherical wrist from the link frame {4} to {5} contains the parameter d_5, which becomes to zero for spherical wrist, but it is not zero for spherical one in Table 3.1. Here, we extend their work by introducing additional indexes OFF_1 and OFF_2 to handle wrist offset case like as in Table3.2 and 3.3.

Table 3.1 The D-H parameters of the PUMA robot of elbow type with spherical wrist

Joint	θ_i	α_i	a_i	d_i	Joint range
1	q_1	-90	0	0	-160~160
2	q_2	0	a_2	d_2	-225~45
3	q_3	90	a_3	0	-45~225
4	q_4	-90	0	d_4	-110~170
5	q_5	90	0	$0(d_5)$	-100~100
6	q_6	0	0	d_6	-226~266

Table 3.2 Offset index of the PUMA robot of elbow type

Index	Val.	Change of Link parameter	
OFF_1	0	$d_2 \leftarrow d_2$	Spherical wrist
	1	$+ OFF_1 \cdot d_5 C_4$	Non-spherical wrist
OFF_2	0	$a_3 \leftarrow a_3$	Spherical wrist
	1	$- OFF_2 \cdot d_5 S_4$	Non-spherical wrist

Wrist offset affects the geometric configuration indexes of PUMA robot such as ARM, ELBOW and WRIST as shown in table 4. As a result, these configuration indexes cannot be utilized to find inverse kinematics solutions in case of the PUMA robot with a wrist offset.

Table 3.3 Configuration of the elbow type PUMA robot:

Index	Val	Description	Related equations
ARM	+1	Right arm	$d_4S_{23}+(a_3-OFF_2\cdot d_5S_4)C_{23}+a_2C_2<0$
	-1	Left arm	$d_4S_{23}+(a_3-OFF_2\cdot d_5S_4)+a_2C_2>0$
ELBOW	+1	Above arm	$d_4C_3-(a_3-OFF_2\cdot d_5S_4)\cdot S_3>0$
	-1	Below arm	$d_4C_3-(a_3-OFF_2\cdot d_5S_4)\cdot S_3<0$
WRIST	+1	Wrist down	$S_5>0$
	-1	Wrist up	$S_5<0$

3.2. Singular Configuration

The Jacobian matrix provides us useful information such as analysis of workspace and singular configuration. Some inverse kinematics methods use Jacobian and its inverse form. The Jacobian matrix J maps the velocity $\dot{q}$ in joint coordinate into the velocity $\dot{x}$ in Cartesian coordinate such that,

$$\dot{x}=J(q)\dot{q} \tag{3.1}$$

In this paper, we derived the closed form of the Jacobian determinant of general six axes robots with wrist offsets for the first time (The details derivation requires very tedious computation more than twenty pages, therefore they will be omitted here but they can be obtained from the author).

$$\begin{aligned}Det(J)&=a_2S_5(d_4S_{23}+a_3C_{23}+a_2C_2)(d_4C_3-a_3S_3)+\\ &d_5a_2S_4[(S_3S_5-C_3C_4C_5)(d_4S_{23}+a_3C_{23}+a_2C_2)+\\ &S_{23}C_4C_5(d_4C_3-a_3S_3)-d_2C_2S_4C_5]\end{aligned} \tag{3.2}$$

From the above equation, we can derive the following Jacobian determinant of the original PUMA robot with a spherical wrist

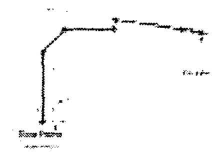
$$Det(J)=a_2(d_4S_{23}+a_3C_{23}+a_2C_2)(d_4C_3-a_3S_3)S_5 \tag{3.3}$$

Note that Eq.(3.3) corresponds to a special case of Eq.(3.2) and it consists of the multiplication of four terms. Here, the 1st term a_2 is non zero constant. If the other terms becomes zero such as $d_4S_{23}+a_3C_{23}+a_2C_2$, $d_4C_3-a_3S_3$ and S_5, then the robot is located in the corresponding singular configuration as shown in Fig. 3.3.

Fig. 3.2(a) occurs when the wrist is on the plane made by the 1st joint and the 2nd one, (b) means that the wrist is on the straight line of the 2nd joint and the 3rd one, and finally (c) is only related to the change of wrist orientation comparing with (b).

However, it is difficult to identify generally the singular configuration of six axes robots with wrist offsets, because Eq.(3.2) can not be expressed by the form Eq.(3.3). This fact makes it is not easy for general robot to identify and avoid singular configurations.

(a) $d_4S_{23}+a_3C_{23}+a_2C_2=0$ (b) $d_4C_3-a_3S_3=0$



(c) $S_5=0$

Fig. 3.2 Singular configurations of PUMA robot:

4. TWO CONVENTIONAL INVERSE KINEMATICS ALGORITHMS

4.1. Conventional Method I

We illustrate a conventional iterative algorithm to find the inverse kinematics of the robot in the US patent [3. Fig.4.1 shows its kinematics graphically.

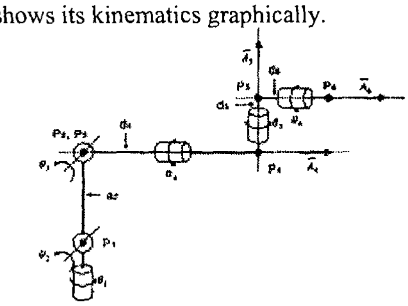


Fig. 4.1 Robot kinematics (US patent[3)

The main procedure of this patent can be summarized as follows.

[Step 1] If the position and orientation are given, $\bar{A}_6$ is constant and P_5 is obtained from $P_5=P_6-d_6\bar{A}_6$

[Step 2] From $\theta_1,\theta_2,\theta_3$, compute $\bar{A}_4$

[Step 3] Compute $\bar{A}_5=\bar{A}_4\times\bar{A}_6$ and $P_4=P_5-d_5\cdot\bar{A}_5$

[Step 4] Compute $\theta_1,\theta_2,\theta_3$ from P_4

[Step 5] Using the $\theta_1,\theta_2,\theta_3$ from step 4 as initial points, repeat the procedure from step 2 to step 4 until the solution satisfying the termination criteria is obtained.

In this method, if the joint variables $\theta_1,\theta_2,\theta_3$ are obtained, they define unique value of θ_4 such that

$$\theta_4=g(\theta_1,\theta_2,\theta_3) \tag{4.1}$$

Step 3 and 4 mean to find the functions $f_i(\bullet), i=1,2.3$ expressed as

$$\theta_1=f_1(\theta_4),\theta_2=f_2(\theta_4),\theta_3=f_3(\theta_4) \tag{4.2}$$

This method has two drawbacks. First it cannot be generally applied to the robots with various type wrist offsets. Second, if two or more values of $\theta_1,\theta_2,\theta_3$ exist during iteration and are not unique, there is no appropriate guide to select the appropriate ones.

4.2. Conventional method II

Doty has proposed the following inverse kinematics algorithm based on one-dimensional Newton-Raphson method[4]. The brief procedure of their method is as follows.

[Step 1] For the given eight functional equations of inverse kinematics containing four joint variables of $\theta_1, \theta_2, \theta_3$ and θ_4, first select one joint variable such as θ_2 and derive the implicit function $f(\theta_2)$ satisfying $f(\theta_2) = 0$

[Step 2] Initialize θ_2

[Step 3] Compute θ_2 with

$$\theta_2 \leftarrow \theta_2 - \frac{f(\theta_2)}{df(\theta_2)/d\theta_2} \tag{4.3}$$

[Step 4] Perform Newton-Raphson iteration by repeating the procedure from Step 2 to Step 3 until the termination criteria is satisfied.

[Step 5] Since the solution of θ_2 is obtained, compute the other joint variables $\theta_1, \theta_3, \theta_4$ using it.

This method has some inherent shortcomings because it is based on Newton-Raphson method, which is likely to fall into non-convergent cycle when it saddles on local optimal points. Thus, it does not guarantee to find the real solution and it requires lots of trials and iterations so as to avoid these local points.

4.3. Fixed point algorithm

Let's revisit the conventional method I. We can restate it that as follows. After we firstly find θ_4 using $\theta_4 = g(\theta_1, \theta_2, \theta_3)$, then we find the initial values of $\theta_1, \theta_2, \theta_3$ satisfying $\theta_i = f_i(\theta_4), i = 1,2,3$. After this, we modify θ_4 using the obtained initial value $\theta_1, \theta_2, \theta_3$. This procedure is repeated until the satisfactory solutions are obtained. In other words, the above algorithm can be expressed as the following equation called fixed-point iteration.

$$\theta_{4,n+1} = g(f_1(\theta_{4,n}), f_2(\theta_{4,n}), f_3(\theta_{4,n})) \tag{4.4}$$

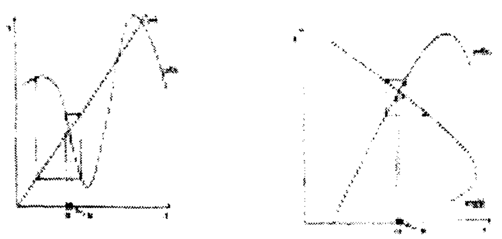
Here, it is necessary to mention that the fixed-point algorithm has its own inherent limitations.

[Theorem]: Convergence of fixed-point algorithm

Let x_s be the unique root solution satisfying $x = g(x)$, and assume a function g is continuous and differentiable in the region D containing x_s. Then, so that the iterative equation $x_{n+1} = g(x_n)$ converges to one root solution, the following condition should be satisfied.

$$|g'(x)| \le K < 1 \quad \text{for} \quad x \in D \tag{4.5}$$

The conventional method I can only find a solution when it satisfies the above theorem. Therefore, this fact means that it has difficulty in most of nonlinear equations. Fig.4.3 shows two illustrations for this problem. We cannot find the root solution x_s using the pure fixed -point algorithm although we put the initial value of x_d is very near to it.



(a) $x_{n+1} = f(x_n)$ (b) $y_n = f(x_n), x_{n+1} = g(x_n)$

Fig. 4.3. Fixed-point iterations:

5. THE SUGGESTED ALGORITHMS

4.1. Main Features

The conventional methods we have mentioned also have another problem. They do not consider the feasible region of solution, thus it often falls into an unfeasible point. The system can be broken down during iterative computational procedure.

In this paper, so as to solve this problem, we propose a unidirectional search algorithm considering the feasible range of solution. Our algorithm can not only overcome these shortcomings of the conventional one using the suggested reflection methods on the boundary and curved points but also it guarantees to find the solution.

Our algorithm not only can find out the exact solution always without errors but also converges very rapidly within ten iterations comparing with conventional methods.

4.2. Search for Feasible Region

Through some tedious procedure, we derived the equations determine the minimum and maximum value of four joint angles $\theta_1, \theta_2, \theta_3, \theta_4$ of PUMA robot. The feasible region that θ_4 can have real value can be obtained by satisfying the following two equations simultaneously

$$p_x^2 + p_y^2 - (d_5 C_4 + d_2)^2 > 0 \tag{4.6}$$

and

$$p_x^2 + p_y^2 + p_z^2 - (d_5 C_4 + d_2)^2 - \left[\frac{p_x^2 + p_y^2 + p_z^2 - (d_5 C_4 + d_2)^2 + a_2^2 - d_4^2 - (d_5 S_4 + a_3)^2}{2a_2}\right]^2 > 0 \tag{4.7}$$

Then, the other joint variables $\theta_1, \theta_2, \theta_3$ can have real valued solution satisfying feasible range.

4.3. Unidirectional Search

The most of iterative algorithms are based on Newton based search. However, they cannot find the solution like as in the two cases of Fig. 4.4.

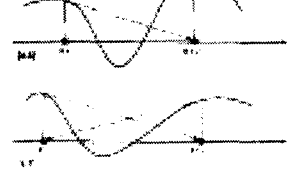


Fig. 4.4 Two unfortunate cases of Newton-Raphson

The upper case of the Fig. 4.4 corresponds to the saddling case of the minimum or maximum, and the lower one denotes to the saddling state in non-convergent cycle during iteration, respectively.

We can see that the both of two cases in Fig.4.4 satisfy the following equation.

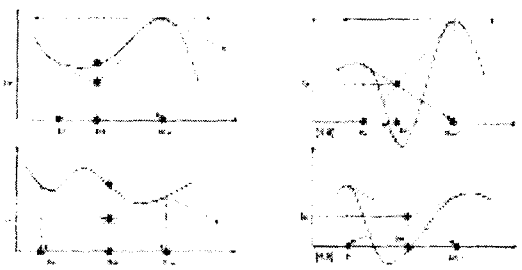
$$f'(x_n)\cdot f'(x_{n+1})\le 0 \tag{4.8}$$

We suggest a unidirectional algorithm to avoid the problems of Newton-Raphson method. It can be described as follows. First, if Eq.(4.8) is satisfied, then update $x_{n+1}, x_m, f(x_m)$ like as in Fig. 4.5(a) using the following equations.

$$x_{n+1} = x_n - f(x_n)/(\frac{\partial f(x_n)}{\partial(x_n)}) \tag{4.9}$$

$$x_m = x_n - \frac{f(x_n)}{\frac{f(x_{n+1})}{x_{n+1}-x_n} - \frac{\partial f(x_n)}{\partial(x_n)}} \tag{4.10}$$

$$f_m = \frac{f(x_n)f(x_{n+1})}{f(x_{n+1}) - \frac{\partial f(x_n)}{\partial(x_n)}(x_{n+1}-x_n)} \tag{4.11}$$



(a)update $x_{n+1}, x_m, f(x_m)$ (b) update x_{n+1} when sign($f(x_m)-f_m$)= sign($f(x_n)$)

Fig. 4.5 Unidirectional search:

Second, compare the sign of $f(x_m)-f_m$ with that of $f(x_n)$, and take different update rules of x_{n+1} as follows. If the sign is the same, then perform,

$$x_{n+1} = x_n - f(x_n)/(\frac{\partial f(x_{n+1})}{\partial(x_n)}) \tag{4.12}$$

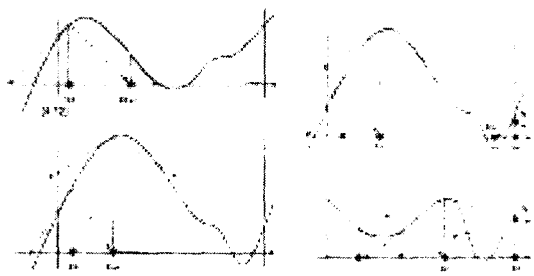
Otherwise,

$$x_{n+1} = x_m - f(x_n)/(\frac{\partial f(x_m)}{\partial(x_n)}) \tag{4.13}$$

Fig. 4.5 (b) shows the unidirectional search of Eq.(4.12) when the sign of $f(x_m)-f_m$ is the same with that of $f(x_n)$.

4.4. Reflection on Curve and Boundary

The inverse kinematics problem of 6 axes robot with wrist offset results in algebraic equations of joint variables $\theta_1, \theta_2, \theta_3, \theta_4$. Fig.4.6 illustrates the cases when the estimated solution based upon Newton-Raphson method is out of feasible range.



(a) Reflection on a curve, (b) Reflection on boundary

Fig. 4.6 Reflection method:

As mentioned previously, joint variable θ_4 should be in real valued range to control real robots. So as to achieve the feasible solutions, we use here a reflection method on the curve to prevent the solution from exceeding its boundary. Since the sign of $f(x_n)$ changes on the curve due to reflection, x_{n+1} is determined as

$$x_{n+1} = x_n + \frac{f(x_n)}{\partial f(x_n)/\partial x_n} \tag{4.14}$$

On the other hand, when the point is exceeded out of the other boundary, then it may be relocated on the original point using only the above method. In this case, we can solve this problem by applying reflection method on the boundary point as shown in Fig.4.6 (b). Then, f_b and x_{n+1} are computed as follows.

$$f_b = f(x_n) - \frac{\partial f(x_n)}{\partial x_n}(x_b - x_n) \tag{4.15}$$

$$x_{n+1} = x_b - \frac{f_b}{\partial f(x_n)/\partial x_n} \tag{4.16}$$

4.5. Overall Flowchart

The overall procedure of the suggested algorithm is shown in Fig. 4.7. Here, SFlag is initially set to 1, otherwise it is set to 0. BFlag is set to 1 only when boundary reflection is permitted, otherwise it is set to 0. Although our algorithm needs three conditional checks and Move computation per each iteration step, it has strong advantage that it can rapidly avoid the local minima point with less computation comparing Newton-Raphson method. Furthermore, the search converges to the global optimal point without saddling on the local point unlike Newton-Raphson method.

6. EXPERIMENTAL RESULTS

The computer simulation was performed to investigate the performance of the proposed inverse kinematics algorithm for the six axes PUMA robot assuming a wrist offset.

Fig. 18 (a) is one of iteration results using the suggested algorithm. Here, you can find that it converged to the exact solution without errors very rapidly within ten iterations Compare (a) with (b),(c) and (d), which are the iteration results of conventional method I in US patent for the same test condition. You can observe that the conventional method I does not converge to the solution unlike our method in this case.

Fig. 18 (e) and (f) show the iteration results of our methods. Compare these with (g) and (h) of Doty's method for the same condition. Here, you can see that our method is faster than Doty's method.

Table 4.1 shows the summarized data of three methods, when the error between the real value of solution and the RMS estimated value exceeds 0.001.

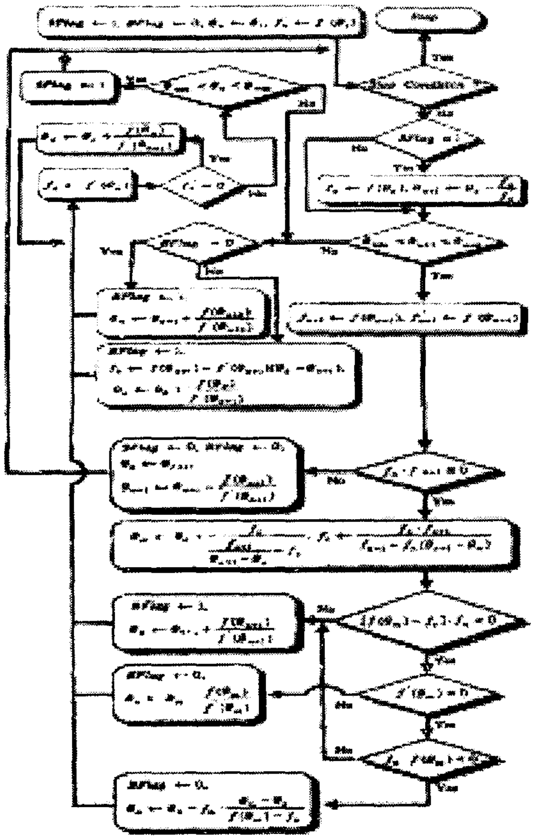


Fig. 4.7 Overall flowchart of the suggested algorithm

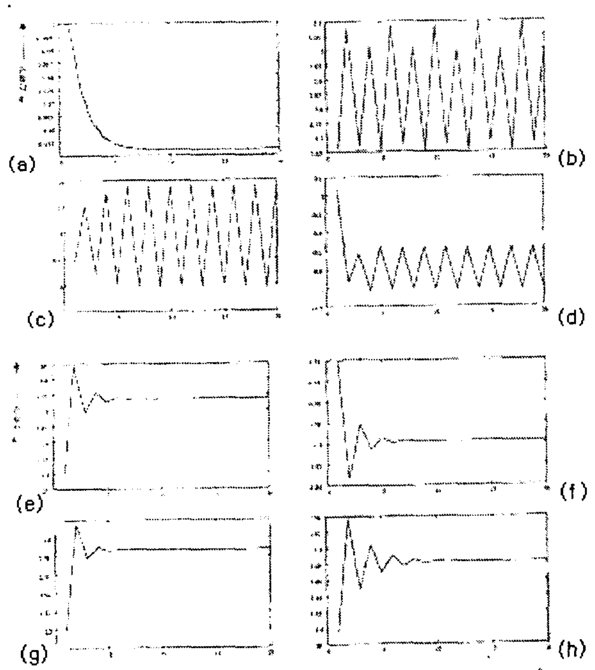


Fig. 18. Simulation results

Table 4.1 Number of errors: real value-estimated value > 0.001

Trial No.	US patent	Doty's	Suggested
10	6	2	0
100	77	25	0
1000	749	293	0
10000	6995	2884	0
100000	73975	27432	0

Through the experiments, the suggested algorithm successfully found the solutions, while the conventional two methods I, II generated numerical errors in some cases. Moreover, they converged slowly to solutions comparing with our method. This is very reasonable and natural because our algorithm has inherently employed the reflection methods to remove error possibilities unlike the conventional methods.

7. CONCLUSION

We will conclude our paper with the following remarks.

First, we considered the feasible inverse kinematics solution assuming that the PUMA robot has a wrist offset and derived Jacobian matrix determinant of PUMA robot with wrist offset case.

Second, we investigated some drawbacks of conventional numerical methods of inverse kinematics solution. They do not consider the feasible regions of solutions and likely to fall into local points.

Third, we proposed a unidirectional search algorithm considering the feasible range of inverse kinematics solution so as to solve the problems of conventional methods.

Finally, our algorithm can not only overcome these shortcomings of the conventional ones but also guarantee to find the exact solution very rapidly.

Thus, our proposed algorithm showed better performance comparing with the conventional methods.

References

[1] J. J. Craig, *Introduction of Robotics: Mechanics and Control*, Addison Wesley Publishing Company, 1989.

[2] C. S. G. Lee and M. Ziegler, "Geometric Approach in Solving Inverse Kinematics of PUMA Robots", IEEE Trans. on Aerospace and Electronic Systems, Vol. AES-20, No. 6, pp. 87-97, 1984

[3] Perzley et al:, "Coordinate Conversion System and Method for Controlling and Industrial Robot," *United System Patent*, Patent Number 4,823,279 (Date of Patent April, 18, 1989)

[4] Rachid Manseur, Keith L. Doty, "A Fast Algorithm for Inverse Kinematic Analysis of Robot Manipulators", *Int. J. of Robotics Research*, Vol.7, No.3, June, 1988

[5] A.G. Erdman, *Modern Kinematics : Developments in the Last Forty Years*, John Wiley & Sons, Inc., 1993.

[6] K. N. Low and R. N. Dubey, A Comparative Study of Generalized Coordinates for Solving the Inverse-Kinematics Problem of a 6R Robot Manipulator", Int. J. of Robotics Research, Vol.5, No.4.

[7] J. Duffy, *Analysis of Mechanism and Robot Manipulators*, John Wiley & Sons, New York, 1980.