Human Arm Inverse Kinematic Solution Based Geometric Relations and Optimization Algorithm

Mohammed Z. Al-Faiz, , MIEEE *mzalfaiz@ieee.org*
omputer Engineering Dept. .
Nahrain University
Baghdad, Iraq

Abduladhem A.Ali *Abduladem1@yahoo.com*
Computer Engineering Dept. .
Basrah University
Basrah, Iraq

Abbas H.Miry *abbasmiry83@yahoo.com*
Electrical Engineering Dept. .
AL-Mustansiriyah University
Baghdad, Iraq

Abstract

Kinematics for robotic systems with many degrees of freedom (DOF) and high redundancy are still an open issue. Namely, computation time in robotic applications is often too high to reach good solution, for parts of the kinematic chain; the problem of inverse kinematics is not linear, as rotations are involved. This means that analytical solutions are only available in limited situations. In all other cases, alternative methods will have to be an employed.The most-used alternative is numerical solutions optimization. This paper presents a strategy based on combine's analytical solutions with nonlinear optimization algorithm solutions to solution the IKP. A analytical solutions is used to reduce the size of problem from seven variable of joint angle to single variable and nonlinear optimization algorithm was used to find approximate solution which make the computation time is very small

Keywords: Inverse Kinematic, Human Arm, Levenbrge Marquite.

1. INTRODUCTION

Kinematics is the study of motion without regard to the forces that create it. The representation of the robot's end-effecter position and orientation through the geometries of robots (joint and link parameters) are called forward Kinematics.[1].The forward kinematics is a set of equations that calculates the position and orientation of the end-effector in terms of given joint angles. This set of equations is generated by using the D-H parameters obtained from the frame assignation. [2].The inverse kinematics problem (IKP) for a robotic manipulator involves obtaining the required manipulator joint values for a given desired end-point position and orientation. It is usually complex due to lack of a unique solution and closed-form direct expression for the inverse kinematics mapping. [3].

Forward kinematics can be formulated for all serial manipulators. However, with increasing degrees-of-freedom (DOF), these solutions become extremely complex and possibly computationally inefficient. Inverse position solutions are however only possible for non-redundant robots that have a finite number of solutions. For redundant systems, inverse kinematics leads to an infinite number of solutions and numerical approaches are the best to be used. Additionally, decision-making and optimization becomes important for redundant system inverse kinematics [4].

Kinematics of the human body is concerned with formulating and solving for the translational and rotational position, analysis problems for each human body segment of interest, for various real world motions. Forward kinematics calculates the pose (position and orientation) of each human body segment of interest given the joint angles. The forward kinematics is the problem of finding an end-effector or tool pose from a set of given joint angles. Inverse kinematics calculates the required joint angles given the current human body (or portion thereof) pose. Statics requires the positions and angles of each segment for static free-body diagrams[1].

Several inverse kinematics algorithms have been proposed. The latest approach dealing with inverse kinematics using nonlinear optimization solution for IKP by Sugihara [5]he use Levenberg-Marquardt method with robust damping with n variable depend on problem which may take long time .

In order to describe a kinematic chain, we are going to view the rotations performed by the joints of a chain separately from the setup of the chain, i.e. length of links,position and orientation of joints, etc , In the space, translation and rotation can be expressed by homogeneous transformation matrices. In the 3D Euclidean

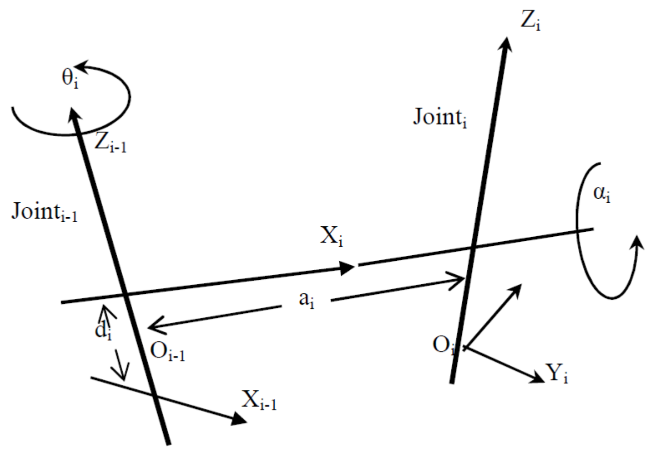

FIGURE 1. The relation between two consecutive coordinates

space, these take the form of 4 × 4 matrices which can be concatenated simply by matrix multiplication. Since it is essential to combine the frame transitions with other translations and rotations, foremost the rotations performed by the joints, the rotation matrices and translation vectors have to be transformed to homogeneous transformation matrices To transform a 3 × 3 rotation matrix or a 3 × 1 translation vector into homogeneous transformation matrices, they are positioned in a 4 × 4 matrix where the remaining entries are filled with the identity matrix. A rotation matrix R replaces the upper left part of the 4 × 4 matrix, a translation vector T replaces the three upper entries of the last column[6]:

$$\left.\begin{aligned} rot_{hom} &= \begin{bmatrix} & R_{3X3} & & \begin{matrix}0\\0\\0\end{matrix} \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ trans_{hom} &= \begin{bmatrix} 1 & 0 & 0 & \\ 0 & 1 & 0 & t_{3X1} \\ 0 & 0 & 1 & \\ 0 & 0 & 0 & 1 \end{bmatrix} \\ rot_{hom}\,trans_{hom} &= \begin{bmatrix} & R_{3X3} & & t_{3X1} \\ 0 & 0 & 0 & 1 \end{bmatrix} \end{aligned}\right\} \qquad (1)$$

As demonstrated, a single homogeneous transformation matrix can be equipped with both a translational and a rotational part. This enables a composition of the rotations performed by the joints and the frame transitions caused by the nature of the kinematic chain.

$$A_i = Rot_{x,\alpha_i}\, Trans_{x,a_i}\, Rot_{z,\theta_i}\, Trans_{z,d_i}$$

$$= \begin{bmatrix} \cos\theta_i & -\sin\theta_i & 0 & b_i \\ \cos\alpha_i \sin\theta_i & \cos\alpha_i \cos\theta_i & -\sin\alpha_i & -d_i \sin\alpha_i \\ \sin\alpha_i \sin\theta_i & \sin\alpha_i \cos\theta_i & \cos\alpha_i & d_i \cos\alpha_i \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (2)$$

The kinematic analysis of an n-link manipulator can be extremely complex and the conventions introduced below simplify the analysis considerably. Moreover, they give rise to a universal language with which robot engineers can communicate. A commonly used convention for selecting frames of reference in robotic applications is the Denavit-Hartenberg, (DH) convention. In this convention, each homogeneous transformation A_i is represented as a product of four basic transformations[7].where the four quantities θ_i α_i, a_i, d_i, are parameters associated with link (*i*) and joint (*i*). The four parameters a_i, α_i, d_i, and θ_i in equation(2) are generally given the names link length, link twist, link offset, and joint angle, These names are derive from specific aspects of the geometric relationship between two coordinate frames, as will become apparent below. Since the matrix A_i is a function of a single variable, it turns out that three of the above four quantities are constant for a given link, while the fourth parameter, θ_i for a revolute joint and d_i for a prismatic joint, is the joint variable.

In order to obtain a systematic representation of the workspace produced by the motion of a point of interest (typically called a point on the end-effector), we will use the (D-H) method adopted from the field of robotics [8].

Let Z_{i-1} and Z_i represent fixed axes at either end of link *i*-1, about which or along which links *i*-1 and *i* move, respectively. Let axes X_{i-1} be defined from Z_{i-1} to Z_i and perpendicular to both. Let Y_{i-1} represent the unique axis that together with X_{i-1} and Z_{i-1} completes a right-hand Cartesian coordinate system. Let Z_i represent a vector from O_{i-1} parallel to Z_i . Let X_{i-1} a vector from O_i parallel to X_{i-1} as illustrated in Fig.1.

Note that the four parameters θ_i α_i, a_i, d_i, completely define the relation between any two consecutive frames. These values are entered in a table, which is typically known as the DH Table. The overall Denavit-Hartenberg respectively coordinate transformation matrix from frame *i* coordinate system relative to the frame *i-1* coordinate system is then given by matrix A_i same as equation(2).

From above, it is clear that the position and orientation of the manipulator end effecter are obtained based on the joint displacements. The joint displacement corresponding to a given end effecter location is obtained by solving the inverse kinematics equations. Hence here, we are concerned not only with the final position of the end effecter, but also with the velocity with which the end effecter moves. describes the kinematics of the manipulator and it maps the joint vector into the end effecter position vector.

2. STRUCURE AND KINEMATIC OF HUMAN ARM

The development of a high-DOF, kinematic is discusses human model that can be used to predict realistic human arm postures. one may deal with anthropomorphic arm by 7-DOF and assume the origin at shoulder joint. The first joint is the shoulder joint s with 3 DOFs. The elbow joint e has only one DOF. The wrist joint w is of the same type as the shoulder joint s and also has 3 DOFs.

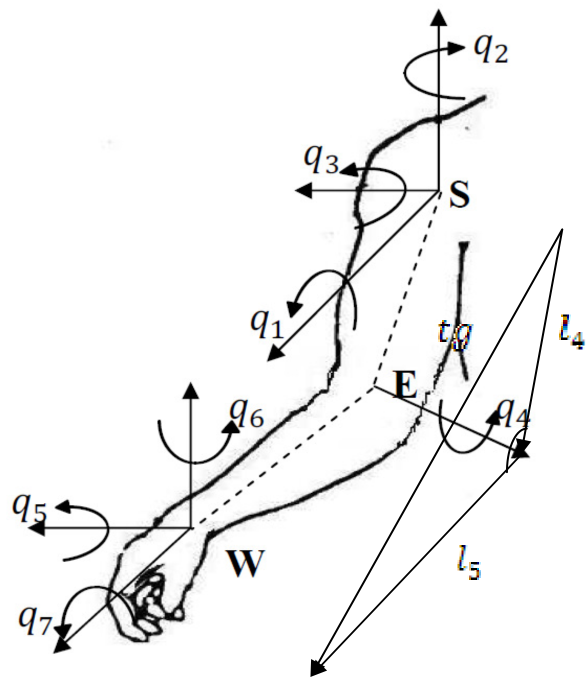

FIGURE 2. Kinematic chain of human arm

Note that the arrow at the end of the chain indicates the end effectors orientation and is not another link. It can be focused on a kinematic chain that is formed after a human arm. This means the kinematic chain has 3 joints with spherical joints as shoulder and wrist joint and a hinge joint as the elbow joint. The spherical joints have 3 DOFS while the hinge joint has only one DOF, giving a total of 7 DOFs for this kinematic chain, see Fig.2. The homogeneous transformation matrices for the frame transitions are set up with D-H parameters [9].

TABLE.1 Numeric Value for D-H Parameters

Frame (joint)	q_i(rad)	d_i(cm)	a_i(cm)	α_i(rad)
1	q_1	0	0	π/2
2	q_2+ π/2	0	0	π/2
3	q_3	0	0	-π/2
4	q_4	0	L4	π/2
5	q_5	0	L5	-π/2
6	q_6- π/2	0	0	-π/2
7	q_7	0	0	-π/2

$$A_1 = \begin{bmatrix} c_1 & -s_1 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ s_1 & c_1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, A_2 = \begin{bmatrix} c_2 & -s_2 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ s_2 & c_2 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, A_3 = \begin{bmatrix} c_3 & -s_3 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -s_3 & -c_3 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_4 = \begin{bmatrix} c_4 & -s_4 & 0 & l_4 \\ 0 & 0 & -1 & 0 \\ s_4 & c_4 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, A_5 = \begin{bmatrix} c_5 & -s_5 & 0 & l_5 \\ 0 & 0 & 1 & 0 \\ -s_5 & -c_5 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}, A_6 = \begin{bmatrix} c_6 & -s_6 & 0 & 0 \\ 0 & 0 & -1 & 0 \\ s_6 & c_6 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$A_7 = \begin{bmatrix} c_7 & -s_7 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ -s_7 & -c_7 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

It has following transformation matrices where 'c' is cosine of theta and 's' is sin of theta, The forward kinematic represents by T

$$T = A_1 * A_2 * A_3 * A_4 * A_5 * A_6 * A_7$$

$$T = \begin{bmatrix} nx & ox & ax & dx \\ ny & oy & ay & dy \\ nz & oz & az & dz \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad (3)$$

3. PROPSED METHOD OF INVERSE KINEMATICS

3.1 Analytical Section

The IK will be considered now. The proposed algorithm can be explain in the following:
Step one: In first we find θ_4 because it depend on posture of arm , tg represent the distance between start point and target point as shown in fig .2 , according to kinematics equation

$tg = \sqrt{(dx^2 + dy^2 + dz^2)}$ (4)

and from the fig .2

$tg = l_4{}^2 + l_5{}^2 - 2l_4 l_5 cos\theta_4$ (5)

Then $\theta_4 = cos^{-1} \frac{tg^2 - l_4{}^2 - l_5{}^2}{2l_4 l_5}$ (6)

Step 2: In this step find θ_1: dx, dz have similar form as shown

$dx = -l_5(c_4(s_1 s_3 + c_1 c_3 s_2) + c_1 c_2 s_4) - l_4(s_1 s_3 + c_1 c_3 s_2)$ (7)

$dz = l_5(c_4(c_1 s_3 - c_3 s_1 s_2) - c_2 s_1 s_4) + l_4(c_1 s_3 - c_3 s_1 s_2)$ (8)

Rearrangement dx and dz

$-dx = s_1(l_5 c_4 s_3 + l_4 s_3) + c_1(l_5 c_4 c_3 s_2 + l_5 c_2 s_4 + l_4 c_3 s_2)$

$dz = c_1 (l_5 c_4 s_3 + l_4 s_3) - s_1 (l_5 c_4 c_3 s_2 + l_5 c_2 s_4 + l_4 c_3 s_2)$

$let\ k_1 = (l_5 c_4 s_3 + l_4 s_3)$ (9)

$k_2 = (l_5 c_4 c_3 s_2 + l_5 c_2 s_4 + l_4 c_3 s_2)$ (10)

Sub k_1, k_2 in eq 7 and 8 to get

$s_1\ k_1 + c_1\ k_2 = -dx$ (11)

$c_1\ k_1 - s_1\ k_2 = dz$ (12)

$let\ rcos\varphi = k_1$ (13)

$rsin\varphi = k_2$ (14)

Sub eq 13 and 14 in eq 11 and 12

$r\ sin\theta_1\ cos\varphi + r\ cos\theta_1\ sin\varphi = -dx$ (15)

$r\ cos\theta_1\ cos\varphi - r\ sin\theta_1\ sin\varphi = dz$ (16)

Rearrangement eq 15 and 16

$r\ (sin\theta_1\ cos\varphi + cos\theta_1\ sin\varphi) = -dx$

$r\ (cos\theta_1\ cos\varphi - sin\theta_1\ sin\varphi) = dz$

Using triangular formula of sinusoidal

$r\ sin(\theta_1 + \varphi) = -dx$ (17)

$$r\ cos(\theta_1+\varphi) = dz \quad (18)$$

Divide eq 17 by eq 18 to get

$$\tan(\theta_1+\varphi) = -\frac{dx}{dz} \quad (19)$$

from this derivative we find the value of θ_1 which depended on φ

$$\theta_1 = \begin{cases} tan^{-1}\left(\frac{-dx}{dz}\right) - \varphi & if\ dz \neq 0 \\ -\frac{\pi}{2} - \varphi & if\ dz = 0 \end{cases} \quad (20)$$

Step 3: In this step find θ_3: from eq 17 the value of r is found

$$r = \frac{-dx}{sin(\theta_1+\varphi)} \quad (21)$$

Eq 9 is equal to eq 13

$$l_5 cos\theta_4 sin\theta_3 + l_4 sin\theta_3 = rcos\varphi$$

Therefore θ_3 can be found

$$\theta_3 = sin^{-1}\frac{rcos\varphi}{l_5 cos\theta_4 + l_4} \quad (22)$$

Step 4: In this step find θ_2: By compare dy with k_2

$$dy = l_5(s_2 s_4 - c_2 c_3 c_4) - l_4 c_2 c_3 \quad (23)$$

Rearrangement k_2 and dy

$$s_2(l_5 c_4 c_3 + l_4 c_3) + c_2(l_5 s_4) = k_2 \quad (24)$$

$$-c_2(l_5 c_4 c_3 + l_4 c_3) + s_2(l_5 s_4) = dy \quad (25)$$

Let

$$k_3 = l_5 c_4 c_3 + l_4 c_3 \quad (26)$$

$$k_4 = l_5 s_4 \quad (27)$$

Which all coefficient are specified, sub eq 26 and in 27 in eq 25 and 26

$$s_2 k_3 + c_2 k_4 = k_2 \quad (28)$$

$$-c_2 k_3 + s_2 k_4 = dy \quad (29)$$

Solving eq 28 and 29 using matrix inversion

$$\begin{bmatrix} k_3 & k_4 \\ k_4 & -k_3 \end{bmatrix}\begin{bmatrix} s_2 \\ c_2 \end{bmatrix} = \begin{bmatrix} k_2 \\ dy \end{bmatrix} \quad (30)$$

$$\begin{bmatrix} s_2 \\ c_2 \end{bmatrix} = \begin{bmatrix} k_3 & k_4 \\ k_4 & -k_3 \end{bmatrix}^{-1}\begin{bmatrix} k_2 \\ dy \end{bmatrix} \quad (31)$$

Let

$$\begin{bmatrix} k_5 \\ k_6 \end{bmatrix} = \begin{bmatrix} k_3 & k_4 \\ k_4 & -k_3 \end{bmatrix}^{-1}\begin{bmatrix} k_2 \\ dy \end{bmatrix} \quad (32)$$

Then

$$sin\theta_2 = k_5 \quad (33)$$

$$cos\theta_2 = k_6 \quad (34)$$

From latest equations we get

$$\theta_2 = \begin{cases} tan^{-1}\frac{k_5}{k_6}\ if\ k_6 \neq 0 \\ \frac{\pi}{2}\ if\ k_6 = 0 \end{cases} \quad (35)$$

After these procedure $\theta_1, \theta_2, \theta_3\ and\ \theta_4$ are found

In next steps we find $\theta_5, \theta_6\ and\ \theta_7$, In eq 3 , $A_1 * A_2 * A_3 * A_4$ is specified ,to find $A_5 * A_6 * A_7$ apply

$$A_5 * A_6 * A_7 = [A_1 * A_2 * A_3 * A_4]^{-1} * T \quad (36)$$

Step 5: In this step find θ_6: from eq 37

$$A_5 * A_6 * A_7 = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \quad (37)$$

And from from given DH parameter

$$A_5 * A_6 * A_7 = \begin{bmatrix} c_5 * c_7 * s_6 - s_5 * s_7 & -c_7 * s_5 - c_5 * s_6 * s_7 & c_5 * c_6 \\ -c_6 * c_7 & c_6 * s_7 & s_6 \\ -c_5 * s_7 - c_7 * s_5 * s_6 & s_5 * s_6 * s_7 - c_5 * c_7 & -c_6 * s_5 \end{bmatrix} \quad (38)$$

By comparing eq 38 and eq 39 we get

$sin\theta_6 = r_{23}$ (39)
then
$\theta_6 = \sin^{-1} r_{23}$ (40)
Step 6: In this step find θ_5: by comparing eq 37 and eq 38 we get
$r_{13} = c_5 * c_6$ (41)
$r_{33} = -c_6 * s_5$ (42)
then dived eq 42 by eq 41 to obtain
$\tan \theta_5 = -\frac{r_{33}}{r_{13}}$ (43)

$$\theta_5 = \begin{cases} \tan^{-1} -\frac{r_{33}}{r_{13}} & if \quad r_{13} \neq 0 \\ \frac{\pi}{2} & \text{if} \quad r_{13} = 0 \end{cases} \quad (44)$$

Step 6: In this step find θ_7: by comparing eq 37 and eq 38 we get
$-c_6 * c_7 = r_{21}$ (45)
$c_6 * s_7 = r_{22}$ (46)

$$\theta_7 = \begin{cases} \tan^{-1} -\frac{r_{22}}{r_{21}} & if \quad r_{21} \neq 0 \\ \frac{\pi}{2} & if \quad r_{21} = 0 \end{cases} \quad (47)$$

From previous derivative we make all angle $(\theta_1, \theta_2, \theta_3, \theta_5, \theta_6\ and\ \theta_7)$ in term single angle φ therefore the our problem convert from multivariable to one variable which reduce the time required to find the solution .until last step analytical solution is performed.

3.2 . Optimization Section

Now the non linear optimization solution is performed depend on seven angle which depend on single angle φ therfor the next problem has single varaible .The robot kinematics is mathematically represented by a set of constraints on the joint displacement vector $\theta = [\theta_1 \ \theta_2 \cdots \theta_7]^T$,where *7* is the degree of freedom. A positional constraint is represented as

$p(\theta) = p^d$ (48)

Where
The vector θ depended on the value of φ from provieus procdure
p^d Target position in the space.
$p(\theta)$ Calculated position as function of joint space

For an orientation constraint,
$R(\theta) = R^d$ (49)
R^d Target orientation in the space.
$R(\theta)$ Calculated orientation as function of joint space
In both cases, the residual vector *e(q)* can be defined as

$$e(\theta) = \begin{cases} P^d - P(\theta) \text{ (for a positional constraint)} \\ \alpha(R^d * R(\theta)^T) \text{ (for an orientational constraint)} \end{cases}$$

(50)

Where $\alpha(RT) = \left[\frac{1}{||l||} tan^{-1} \frac{||l||}{r_{11}+r_{22}+r_{33-1}}\right] l$ (51)

$$RT = R^d * R^T(q) = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} \quad (52)$$

$$l = \begin{bmatrix} r_{32} - r_{23} \\ r_{13} - r_{31} \\ r_{12} - r_{12} \end{bmatrix} \quad (53)$$

Our interest starts from solving the following nonlinear
Equation:

e(θ)=0 (54)

The conventional IK based on NR tries to find $\theta = \theta^*$ which satisfies Eq.(54) by the following update rule

$$q^{k+1} = q^k - J^{-1}e(\theta^k) \tag{55}$$

Where

$$J = \nabla e(\theta^k) \tag{56}$$

After introduce both analytic and non linear optimization the IK solution will be considered now. The explanation of solution for the kinematic chain introduced in the previous section is present. It required minimizing the error between the target transformation matrix and calculated transformation matrix ,the problem can be formularizing as optimization problem as following:
Minimize

$$e(\theta_1,\theta_2,\theta_3,\theta_4,\theta_5,\theta_6,\theta_7) = \begin{bmatrix} |P_x^d - dx| \\ |P_y^d - dy| \\ |P_z^d - dz| \\ a \end{bmatrix} \tag{57}$$

4. OPTIMIZATION PROBLEM

For optimization of a reference joint angle configuration regarding the similarity measure, one can use the Levenberg-Marquardt algorithm. The algorithm, which was first introduced in [10], provides a standard technique for solving nonlinear least squares problems by iteratively converging to a minimum of function expressed as sum of squares. Combinating the Gauss-Newton and the steepest descent method, the algorithm unites the advantages of both methods. Hence,using the LM method, a more robust convergence behavior is achieved at points far from a local minimum, while a faster convergence is gained close at a minimum. Due to its numerical stability, the LM method has also become a popular tool for solving inverse kinematics problems as demonstrated in [11]

5. LEVENBERG-MARQUARDT ALGORITHM

The Levenberg–Marquardt method uses the second-order derivatives of the mean squared error, so that better convergence behavior is observed [12]. It is assumed that function f (θ) and its Jacobean J are known at point [θ]. The aim of the Levenberg–Marquardt algorithm is to compute the variable vector[θ] such that

$$e(\theta) = d_t - F(\theta) \tag{58}$$

is minimum[13]. It offers an efficient technique that combines regularization with second-order training. It capitalizes on the squared error function

$$E(\theta) = (d_t - F(\theta))^2 \tag{59}$$

Where $F(\theta)$ function of variable and d_t target and uses an efficient approximation to the Hessian [14] the Hessian matrix can be approximated as

$$H = J^T J \tag{60}$$

and the gradient can be computed as

$$g = J^T e \tag{61}$$

The Levenberg–Marquardt algorithm uses this approximation to the Hessian matrix in the following Newton-like update:

$$\theta_{k+1} = \theta_k - [J^T J + \mu I]^{-1} J^T e \tag{62}$$

where J contains first derivatives of the function errors with respect to the variables, I is the identity matrix.When the scalar μ is zero, this is just Newton's method, using the approximate Hessian matrix. When μ is large, this becomes the gradient descent with a small step size. Newton's method is faster and more accurate near an error minimum, so the goal is to shift toward Newton's method as quickly as possible. Thus, μ is decreased after each successful step (reduction in performance function) and is increased only when a tentative step increases the performance function. In this way, the performance function will always be reduced at each iteration of the algorithm[15].In this paper Eq (62) replaced with

$$\theta_{k+1} = \theta_k - [V[S^T S + (\lambda + c)I]V^T]^{-1} J^T e \quad (63)$$

Where
V,S singular value decomposition of J .
c constant is chosen to avoidance the singularity problem,.

6 . EXPERIMENTAL RESULTS

Several experiments were conducted to validate the derived inverse kinematics algorithm. An experiment was conducted to check the correctness of the proposed method derived in Section 4, with reachable positions and orientations.
In t experiment y=13 cm and z=23 cm while x is vary from 25 cm to 30 cm with rotation matrix

$$R = \begin{bmatrix} 0 & 0.7071 & -0.7071 \\ 0 & -0.7071 & -0.7071 \\ -1 & 0 & 0 \end{bmatrix}$$

Table 2 shows the norm error and the computation time in proposed method , LM Algorithm and Sugihara method , experiment results show that the derived inverse kinematics provide lower error with minimum time with respect to other method because the optimization problem with one variable as shown solutions in this experiment.

TABLE. 2 Computation Time and the Error of simulation result of the targets for experiment result with x vary from 25 to 30

x	Error performance index e(Q)			Computation Time (sec)		
	LM Algorithm	Sugihara Method	proposed Method	LM Algorithm	Sugihara Method	proposed Method
25	0.6694	0.6692	0.0005	0.0585	0.0992	0.0139
26	0.6801	0.6801	0.0004	0.0505	0.0995	0.0093
27	0.6937	0.6933	0.0004	0.0515	0.0986	0.0111
28	0.7091	0.7091	0.0003	0.0512	0.0983	0.0115
29	0.7274	0.7272	0.0003	0.0504	0.1000	0.0141
30	0.7477	0.7474	0.0003	0.0516	0.0985	0.0118

Through the simulation it notice that, the error of LM Algorithm and Sugihara method increased with the time because the error at each step will effect on the error of next step ,this will make the error will be accumulative. In the proposed method the difference between actual joints angles and their desired approximately zero, this lead to make the error will not increase with the time. The reason behind the good result of proposed method that, it was near to the perfect solution due to the derivation of equation of human arm and take the advantage of the geometric relations between the equation of human arm. Therefore to find the desired angle only a small displacement in this geometric relations to get the required angles, this make the proposed method provide minimum error and lower time with respect to Sugihara method. The computation time of proposed method is lower than other methods due to using single variable in the search instead the seven variables,

CONCULSION

In this paper, a strategy based on combines an analytical solution with nonlinear optimization algorithm solutions was proposed to solution the IKP. A analytical solutions was used to reduce the size of problem from seven variable of joint angle to one variable nonlinear and optimization algorithm was used to find approximate solution .Combining these method can remedy the weakness of each other, by take the advantage of analytical solutions which provides the correct joint angles for manipulation of the arm end-effectors to any given reachable position and orientation with nonlinear optimization method which find the approximate solution when no exact solution is provide. Sufficient and necessary criteria were provided to determine whether correct solutions existed using forward kinematics. The minimization of the distance between the end effecter and a prescribed Cartesian point is the natural constraint which has been used to reach a solution the proposed method for solving the IKP is that it can be extended to any robotic manipulator, given a set of operation space and joint space parameter values. From the simulation it concluded that, the new method is introduced with minimum error and lower computational time are achieved

REFERENCE

[1] V. Grecu, N. Dumitru, and L. Grecu," Analysis of Human Arm Joints and Extension of the Study to Robot Manipulator", Proceedings of the nternational MultiConference of Engineers and Computer Scientists, Vol 2, March 18 - 20, 2009 pp:1348-1351. .

[2] De Xu, and A. Acosta ," An Analysis of the Inverse Kinematics for a 5-DOF Manipulator" ,International Journal of Automation and Computing vol. 2 ,2005 pp :114-124

[3] C. Hua, C. and Wei-shan, "Wavelet network solution for the inverse kinematics problem in robotic manipulator",Chen Zhejiang Univ Science , Vol. 7 ,No.4 ,2006 PP: 525-529.

[4] Chetan K. and Delbert T. "Kinematics Abstraction for General Manipulator Control", Robotics Research Group, The University of Texas at Austin, JJPRC/MERB 1.206, mail Code R9925, Austin, Texas 78712-1100, 1990 .

[5] T. Sugihara," Solvability-unconcerned Inverse Kinematics based on Levenberg-Marquardt Method", 2009 IEEE-RAS International Conference on Humanoid Robots, , Paris, Dec, 2009 ,pp:555-560

[6] H. Yali and W. Xingsong, "Kinematics analysis of lower extremity exoskeleton", Control and Decision Conference, 2008. CCDC 2008. Chinese, 2-4 July 2008 ,pp: 2837 – 2842 .

[7] M. W. Spong, S. Hutchinson, M. Vidyasagar, "Robot Modeling and Control", JOHN WILEY & SONS, INC, First Edition, 2001, PP:66-67.

[8] K. Abdel-Malek, W. Yu, "Human Placement for Maximum Dexterity", Department of Mechanical Engineering The University of Iowa, Technical report , 2004 pp :1-27.

[9] M. Z. Al-Faiz, Y. I. Al-Mashhadany, "Analytical Solution for Anthropomorphic Limbs Model (IK of Human Arm)", IEEE Symposium on Industrial Electronics and Applications (ISIEA 2009), Malaysia ,October 2009 pp: 684-689.

[10] K. Levenberg, “A Method for the Solution of Certain Non-Linear Problems in Least Squares,” The Quarterly of Applied Mathematics, Vol. 2, 1944 ,pp: 164–168

[11] Do, M.; Azad, P.; Asfour, T.,” Imitation of human motion on a humanoid robot using non-linear optimization”, Humanoid Robots, 2008. 8th IEEE-RAS International Conference pp : 545 – 552 .

[12] Saraoglu, H.M.; Kocan, M. ,”Determination of Blood Glucose Level-Based Breath Analysis by a Quartz Crystal Microbalance Sensor Array”, IEEE Sensor Journal, Vol. 10, No. 1, January 2010 pp: 104-109

[13] E.Derya,” Detecting variabilities of ECG signals by Lyapunov exponents”, Neural Computing & Applications, Springer Vol.18 ,No.7, 2009 pp: 653-662 .

[14] Derrick T. Mirikitani, and Nikolay Nikolaev ,” Recursive Bayesian Recurrent Neural Networks for Time-Series Modeling “IEEE Trans on Neural Network, Vol .21, No. 2, Feb 2010 pp: 262-274.

[15] E. Kolay , K. Kayabali and Y. Tasdemir ,”Modeling the slake durability index using regression analysis,artificial neural networks and adaptive neuro-fuzzy methods”,Bull Eng Geol Environ Springer Vol.69 ,No.7, 2010 pp:275–286.