Efficient kinematics using quaternions for serial and antiparallel link robot arms

Kazutoshi Nishii[1*], Akira Hatano[1] and Yoshihiro Okumatsu[1]

***Abstract*— Dual quaternions have been proposed as an alternative to homogeneous transformation matrices in robotic kinematics. To perform inverse kinematics calculations using quaternions, logarithmic mapping of quaternions is required, however two challenges remain. In states with large errors, the analytical gradient becomes significantly large and stable convergence cannot be obtained. Additionally, there is a tendency to fall into local solutions during the convergence process. Furthermore, efficient modeling methods for parallel-link mechanisms have not yet been discussed. This study proposes a new method for stably obtaining the analytical gradient of quaternions. The proposed method was applied to a robot arm that combined serial and antiparallel link mechanisms, and a new kinematics modeling method for the antiparallel link mechanisms was proposed. The proposed method provides a more stable solution than conventional methods, and an improvement of approximately 10% in solution accuracy was confirmed.**

***Index Terms*— Kinematics, parallel link.**

I. Introduction

Quaternions [1], which express physical phenomena with four elements, do not have the singular configurations of Euler-angle representation. Moreover, the representation of rigid-body transformations using dual quaternions, which involve two quaternions, allows for higher computational efficiency compared to representations such as homogeneous transformation matrices [2], [3], [4]. Their application range is wide, starting with the application to the kinematics of link mechanisms [5] in the old days, and in the field of robotics alone, it spans a wide range of areas such as position feedback control [6], admittance control [7], [8], efficient trajectory generation [9], robust control [10], cooperative control of dual-arm manipulation [11], mobile manipulators [12], SLAM [13], position and attitude control of multi-agent systems [14], cooperative manipulation of multiple robots [15], and cooperative work between humans and robots [16].

This study focuses on kinematics using quaternions, which form the foundation of these fields. Research on kinematics using quaternions has mainly been conducted for serial links, and few studies have been applied to parallel link mechanisms [17], [18]. Even in these studies, the same method as serial link mechanisms was adopted to model each link of the parallel link mechanism, leaving room for the consideration of quaternion representations suitable for parallel link mechanisms.

[1]K. Nishii, A. Hatano and Y. Okumatsu are with the Frontier Research Center, Toyota Motor Corporation, 543 Kirigahora, Nishihirose-cho, Aichi, Japan, e-mail: {kazutoshi_nishii, akira_hatano_aa, yoshihiro_okumatsu}@mail.toyota.co.jp

*Corresponding author

Moreover, kinematics are represented by dual quaternions [19] for rigid-body coordinates, and recent studies have been based on this theory [20], [21], [22]. Ozgur et al.[23] propose a compact representation of dual quaternions using the product of exponentials formula from screw theory. Han et al. [21] demonstrated Lyapunov stability for logarithmic feedback in kinematics using dual quaternions. Although these methods using screw theory and logarithmic mapping are efficient, there remain redundant parts in calculations involving dual number. It is suggested that using two quaternions as a pair of a conventional quaternion and a translation vector can improve computational efficiency [24]. However, these studies warrant additional scrutiny concerning issues associated with the convergence of analytical gradients in inverse kinematics. During the convergence of quaternion errors, the gradient may become zero despite the presence of residual errors, which may result in settling at local optima.

The main contributions of this study are the proposal of an efficient quaternion representation for anti-parallel link mechanisms, which is a type of parallel link mechanism, and an improved method for calculating the analytical gradient in inverse kinematics. Section II discusses the coordinate-transformation representation of the robot link mechanism using a quaternion. We present an efficient model-construction method that focuses on the driving method of a parallel-link mechanism, which is the first of such contribution. In Section III, we outline the kinematics calculation method, identify the analytical gradient issues in the inverse kinematics calculation, and elaborate on the second contribution. In Section IV, we apply the proposed method to an arm that combines serial and parallel-link mechanisms and confirm the effectiveness of our approach.

II. Robot arm modeling

A. Quaternion representation for rigid transformation

The formulation of rigid transformation used in kinematics is represented as a pair of a unit quaternion $\boldsymbol{q_r} \in \mathbb{H}$, which represents posture, and a pure imaginary quaternion $\boldsymbol{q_t} \in \mathbb{H}$ (a quaternion with a zero scalar part), which represents position.

$$\hat{\boldsymbol{\sigma}} = \langle \boldsymbol{q_r}, \boldsymbol{q_t} \rangle .$$

While there is a method to derive these quaternions in the same manner as the DH parameters [20], this study adopts a method to directly derive quaternions from the structural characteristics of the robot [23]. The individual quaternions obtained are expressed as follows, based on a

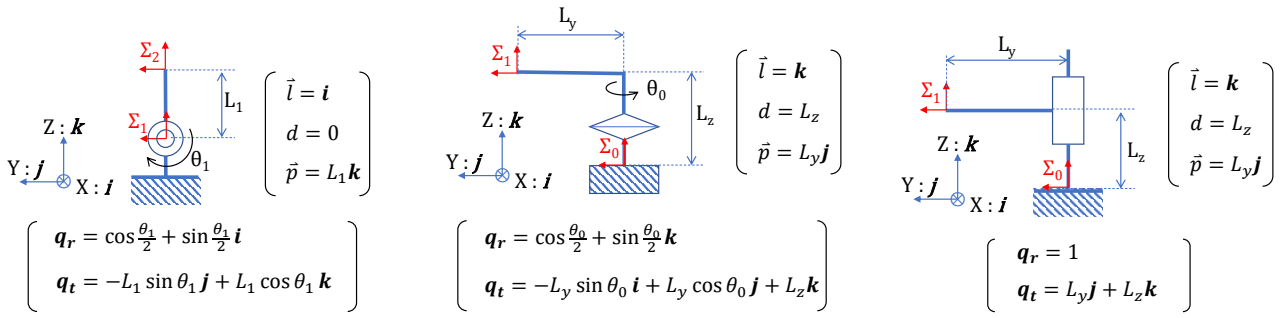


Fig. 1: Example of expressing the coordinate transformation between robot joints using dual quaternions. The correspondence between the local coordinate system (XYZ) and $\boldsymbol{i}, \boldsymbol{j}, \boldsymbol{k}$ is performed based on the reference posture when the joint angle θ is set to zero.

the joint operation angle of θ:

$$\begin{aligned} \boldsymbol{q_r} &= \cos\frac{\theta}{2} + \vec{l}\sin\frac{\theta}{2} \\ \boldsymbol{q_t} &= d\vec{l} + \vec{p}\cos\theta + \vec{l}\times\vec{p}\sin\theta. \end{aligned} \tag{1}$$

Here,

- $\vec{l}$ represents the direction of the rotational axis, which is a complex unit vector.
- d is the distance between coordinates along $\vec{l}$
- $\vec{p}$ represents the shortest three-dimensional complex distance vector from the rotation axis represented by $\vec{l}$ to the coordinates after transformation.

A complex unit vector is defined as the unit vector of a local coordinate system with imaginary units $\boldsymbol{i}, \boldsymbol{j}, \boldsymbol{k}$. They have the following properties as imaginary unit vectors of quaternions.

$$\boldsymbol{ii} = \boldsymbol{jj} = \boldsymbol{kk} = \boldsymbol{ijk} = -1.$$

From the above properties, the multiplication of different unit complex vectors follows the following relationship:

$$\boldsymbol{ij} = \boldsymbol{k}, \boldsymbol{jk} = \boldsymbol{i}, \boldsymbol{ki} = \boldsymbol{j}, \boldsymbol{ji} = -\boldsymbol{k}, \boldsymbol{ik} = -\boldsymbol{j}, \boldsymbol{kj} = -\boldsymbol{i}.$$

This multiplication of different complex unit vectors is the same as the cross-product of vectors when i, j, and k are defined as the following three-dimensional unit vectors:

$$\boldsymbol{i} = \begin{pmatrix} 1 \\ 0 \\ 0 \end{pmatrix}, \boldsymbol{j} = \begin{pmatrix} 0 \\ 1 \\ 0 \end{pmatrix}, \boldsymbol{k} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix}.$$

When considering each parameter in the actual link structure, it is easier to understand it as a three-dimensional vector. Fig.1 shows the link structure specifically related to $\boldsymbol{q_r}, \boldsymbol{q_t}$, and $\vec{l}, d, \vec{p}$.

B. Serial link modeling

In this section, we describe the modeling of a real robot using dual quaternions. The target robot was a 6-degree-of-freedom arm with a parallel link mechanism and a wrist with two degrees of freedom, as shown in Fig.2 [25].

Table I shows the coordinate transformations of the serial link mechanism from the base link of the wrist joint in the base coordinate system Σ_0 to Σ_4. Additionally, the coordinate transformation of the fixed joint from the output

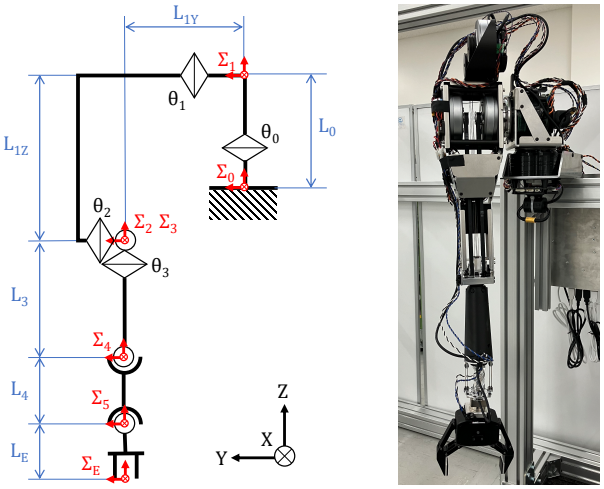


Fig. 2: 6DOF robot arm model

link of the wrist joint to the end effector from Σ_5 to Σ_6 is also given in the same table.

$$\begin{aligned} {}^0\boldsymbol{q_{r1}} &= \cos\frac{\theta_0}{2} + \sin\frac{\theta_0}{2}\boldsymbol{k} \\ {}^0\boldsymbol{q_{t1}} &= L_0\boldsymbol{k} \end{aligned} \tag{2}$$

TABLE I: Serial link coordinate transformations

Coordinate transformation	Quarternions	Link parameters
${}^0\hat{\boldsymbol{\sigma}}_1$	${}^0\boldsymbol{q_{r1}} = \cos\frac{\theta_0}{2} + \sin\frac{\theta_0}{2}\boldsymbol{k}$ ${}^0\boldsymbol{q_{t1}} = L_0\boldsymbol{k}$	$\vec{l}_1 = \boldsymbol{k}$ $d_1 = L_0$ $\vec{p}_1 = \boldsymbol{0}$
${}^1\hat{\boldsymbol{\sigma}}_2$	${}^1\boldsymbol{q_{r2}} = \cos\frac{\theta_1}{2} + \sin\frac{\theta_1}{2}\boldsymbol{j}$ ${}^1\boldsymbol{q_{t2}} = -L_{1Z}\sin\theta_1\boldsymbol{i} + L_{1Y}\boldsymbol{j}$ $-L_{1Z}\cos\theta_1\boldsymbol{k}$	$\vec{l}_2 = \boldsymbol{j}$ $d_2 = L_{1Y}$ $\vec{p}_2 = -L_{1Z}\boldsymbol{k}$
${}^2\hat{\boldsymbol{\sigma}}_3$	${}^2\boldsymbol{q_{r3}} = \cos\frac{\theta_2}{2} + \sin\frac{\theta_2}{2}\boldsymbol{j}$ ${}^2\boldsymbol{q_{t3}} = \boldsymbol{0}$	$\vec{l}_2 = \boldsymbol{j}$ $d_2 = 0$ $\vec{p}_2 = \boldsymbol{0}$
${}^3\hat{\boldsymbol{\sigma}}_4$	${}^3\boldsymbol{q_{r4}} = \cos\frac{\theta_3}{2} + \sin\frac{\theta_3}{2}\boldsymbol{k}$ ${}^3\boldsymbol{q_{t4}} = -L_3\boldsymbol{k}$	$\vec{l}_2 = \boldsymbol{k}$ $d_2 = -L_3$ $\vec{p}_3 = \boldsymbol{0}$
${}^5\hat{\boldsymbol{\sigma}}_E$	${}^5\boldsymbol{q_{r}}_E = 1$ ${}^5\boldsymbol{q_{t}}_E = -L_E\boldsymbol{k}$	$\vec{l}_5 = \boldsymbol{k}$ $d_5 = -L_E$ $\vec{p}_5 = \boldsymbol{0}$

C. Antiparallel link modeling

Next, we model the parallel-link mechanism of the wrist. Various types of parallel-link mechanisms exist, and kinematics are considered for each mechanism [26]. Generally, the base and output links are connected by multiple links, and the position and orientation of the output link are controlled by the length of the connecting links using either a rotational or translational mechanism.

The antiparallel mechanism of the target link is shown in Fig.3. A translational mechanism that changes the length of two pairs of wires was used to provide the output link with two degrees of freedom for tilting.

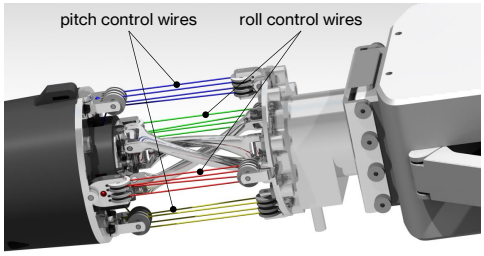


Fig. 3: Wrist CAD model

The three central antiparallel link mechanisms act as follower joints and are two-degree-of-freedom mechanisms that maintain a constant distance between the base and output links. In previous studies [17], [18], the parallel link mechanism was directly replaced using dual quaternions. However, in this study, we simplify the coordinate transformation representation by replacing it with a serial link mechanism with equivalent degrees of freedom. Fig. 4 shows the equivalent model of the parallel-link mechanism targeted in this study, which was replaced by a serial-link mechanism with the same degrees of freedom.

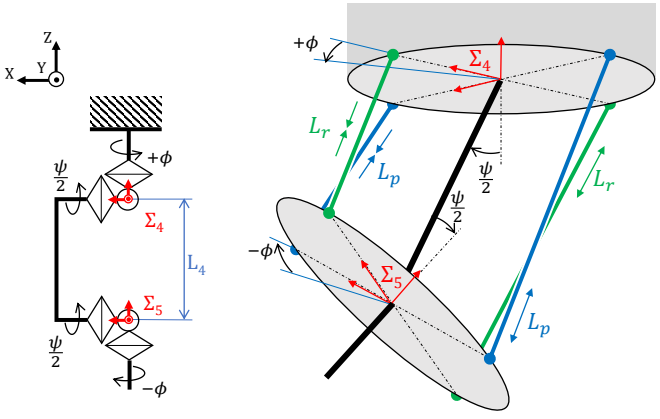


Fig. 4: Wrist link model

The equivalent model has two rotational degrees of freedom around the Z- and X-axes; however, it is constrained by an antiparallel link mechanism. Therefore, if the rotational angles of the two degrees of freedom of the base link are determined, the two degrees of freedom of the motion angles of the output link are uniquely determined. Applying a coordinate transformation using a 4-degree-of-freedom dual quaternion to this equivalent serial link mechanism yields the following results:

$$
\begin{aligned}
{}^4\boldsymbol{q}_{\boldsymbol{r}5} &= \cos\frac{\psi}{2} + \cos\phi\sin\frac{\psi}{2}\boldsymbol{i} + \sin\phi\sin\frac{\psi}{2}\boldsymbol{j} \\
{}^4\boldsymbol{q}_{\boldsymbol{t}5} &= -L_4\Big(\sin\phi\sin\frac{\psi}{2}\boldsymbol{i} - \cos\phi\sin\frac{\psi}{2}\boldsymbol{j} + \cos\frac{\psi}{2}\boldsymbol{k}\Big).
\end{aligned} \tag{3}
$$

We consider the attributes of a wire mechanism that functions as a linear actuator. When the output link deviates from the neutral position by an angle of ψ in the roll direction, the wire length alters by $L_r = 2R_w \sin\frac{\psi}{2}$(Fig. 5a), given that R_w is the radius at which the wire is attached. Subsequently, let us contemplate a scenario where the output link rotates further by an angle of ϕ (Fig. 5b). In this case, the attachment radius of the wire on the roll side becomes $R_w\cos\phi$, the length of the wire on the roll side changes by $L_r = 2R_w\cos\phi\sin\frac{\psi}{2}$. Similarly, it can be postulated that the change in length of the wire on the pitch side amounts to $L_p = 2R_w\sin\phi\sin\frac{\psi}{2}$.

We considered changes in the length of the roll and pitch wires divided by the diameter at which the wires were attached.

$$
\begin{aligned}
q_{wx} &= \frac{L_r}{2R_w} = \cos\phi\sin\frac{\psi}{2} \\
q_{wy} &= \frac{L_p}{2R_w} = \sin\phi\sin\frac{\psi}{2}
\end{aligned}
$$

Introducing the value of $q_{wz} = \cos\frac{\psi}{2}$, (3) can be expressed as

$$
\begin{aligned}
{}^4\boldsymbol{q}_{\boldsymbol{r}5} &= q_{wz} + q_{wx}\boldsymbol{i} + q_{wy}\boldsymbol{j} \\
{}^4\boldsymbol{q}_{\boldsymbol{t}5} &= -L_4\left(q_{wy}\boldsymbol{i} - q_{wx}\boldsymbol{j} + q_{wz}\boldsymbol{k}\right).
\end{aligned}
$$

Because q_{wx}, q_{wy} can be calculated from the wire length, trigonometric operations such as those in Equation (3) are unnecessary. In addition, it is possible to derive q_{wz} without using trigonometric functions, as shown in the following relationship:

$$
q_{wx}^2 + q_{wy}^2 + q_{wz}^2 = 1.
$$

By leveraging this characteristic, the representation through the equivalent model can realize more computationally efficient kinematics compared to the direct replacement of the parallel link mechanism that necessitates trigonometric operations.

III. Quaternion Kinematics

A. Forward kinematics

Let us now consider the transformation from coordinate system Σ_0 to Σ_1 as ${}^0\hat{\boldsymbol{\sigma}}_1 = \langle {}^0\boldsymbol{q}_{\boldsymbol{r}1}, {}^0\boldsymbol{q}_{\boldsymbol{t}1}\rangle$, and the transformation from coordinate system Σ_1 to Σ_2 as ${}^1\hat{\boldsymbol{\sigma}}_2 = \langle {}^1\boldsymbol{q}_{\boldsymbol{r}2}, {}^1\boldsymbol{q}_{\boldsymbol{t}2}\rangle$. At this point, the transformation from coordinate system Σ_0 to Σ_2, denoted as ${}^0\hat{\boldsymbol{\sigma}}_2 = \langle {}^0\boldsymbol{q}_{\boldsymbol{r}2}, {}^0\boldsymbol{q}_{\boldsymbol{t}2}\rangle$, can be represented as follows.

$$
\begin{aligned}
{}^0\boldsymbol{q}_{\boldsymbol{r}2} &= {}^0\boldsymbol{q}_{\boldsymbol{r}1}\,{}^1\boldsymbol{q}_{\boldsymbol{r}2} \\
{}^0\boldsymbol{q}_{\boldsymbol{t}2} &= {}^0\boldsymbol{q}_{\boldsymbol{r}1}\,{}^1\boldsymbol{q}_{\boldsymbol{t}2}\,{}^0\boldsymbol{q}_{\boldsymbol{r}1}^* + {}^0\boldsymbol{q}_{\boldsymbol{t}1}.
\end{aligned} \tag{4}
$$

where, ${}^0\boldsymbol{q}_{\boldsymbol{r}1}^*$ is the complex conjugate of ${}^0\boldsymbol{q}_{\boldsymbol{r}1}$, which is obtained by inverting the sign of the imaginary part of ${}^0\boldsymbol{q}_{\boldsymbol{r}1}$.

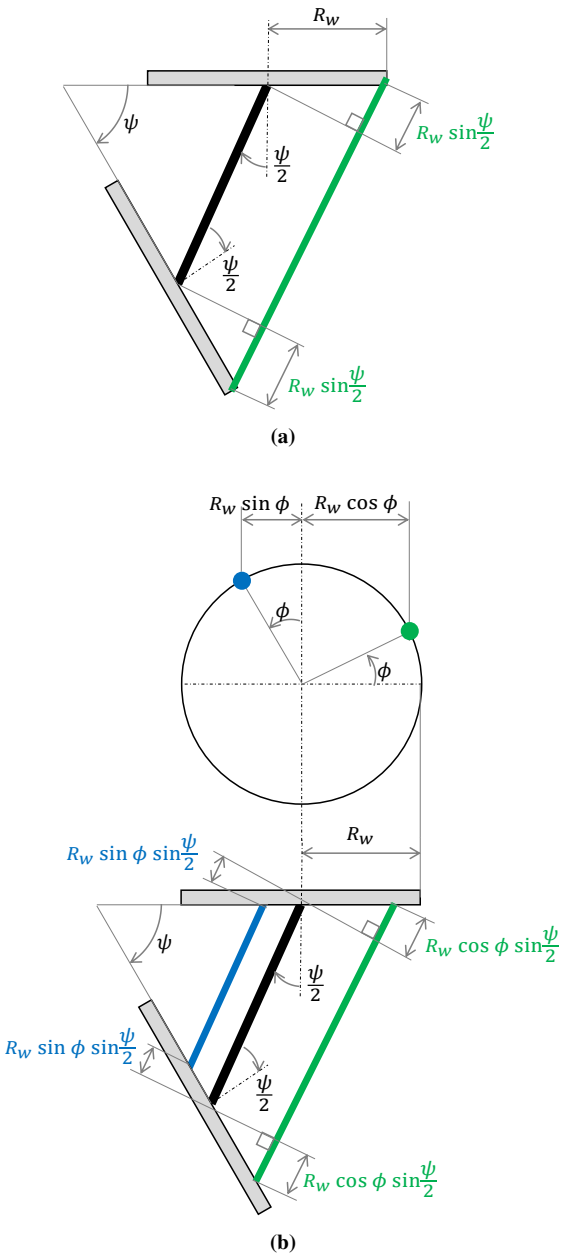


Fig. 5: Wrist wire model

Subsequently, by sequentially performing this operation up to the end-effector coordinate system, the position and orientation of the end effector can be determined.

B. Inverse kinematics

In inverse kinematics, a method is applied that uses the fundamental Jacobian matrix, which represents the relationship between the angles of each joint and the posture and position of the end-effector. Using this method, the update amount of each joint angle is calculated from the posture and position errors of the end-effector and the fundamental Jacobian matrix.

The update amount of the angle $\delta\boldsymbol{\theta_r}$ based on the quaternion of the orientation error is determined by the dot product of the fundamental Jacobian matrix $\boldsymbol{J_r}$, which represents the relationship between the orientation and each joint angle, and the analytical gradient $\delta\boldsymbol{\theta}$.

$$\delta\boldsymbol{\theta_r} = \boldsymbol{J_r} \cdot \delta\boldsymbol{\theta}.$$

$\boldsymbol{J_r}$ is defined as a $N \times 3$ matrix for an arm with N degrees of rotational freedom as follows:

$$\boldsymbol{J_r} = \begin{bmatrix} \vec{r}_0 & \vec{r}_1 & \dots & \vec{r}_{N-1} \end{bmatrix}^T$$
$$\vec{r}_n = {}^0\boldsymbol{q_{r_n}} \vec{l}_n {}^0\boldsymbol{q_{r_n}}^*,$$

where, $\vec{l}_n$ represents the direction of the rotation axis in the nth coordinate system and is a unit complex vector. Additionally, when $n = 0$, because ${}^0\boldsymbol{q_{r_0}}$ does not provide a coordinated rotation operation, $\vec{r}_0 = \vec{l}_0$. If the structure includes a prismatic joint, the corresponding component of $\boldsymbol{J}_r$ becomes zero because there is no effect on the orientation.

The analytical gradient $\delta\boldsymbol{\theta}$ provides a gradient near the error quaternion between the current and the target values. In the case of quaternion orientation, assuming that the target orientation quaternion is $\boldsymbol{q_{r_{ref}}}$ and the current orientation is $\boldsymbol{q_{r_{act}}}$, the error quaternion $\boldsymbol{q_{r_{err}}}$ is given as follows:

$$\boldsymbol{q_{r_{err}}} = \boldsymbol{q_{r_{ref}}} \boldsymbol{q_{r_{act}}}^*. \tag{5}$$

The imaginary part of this quarternion error approaches zero as the orientation error decreases. The analytical gradient was calculated by differentiating the position and orientation with respect to each joint angle. However, in the case of unit quaternions, the four elements are linearly dependent; therefore it is impossible to calculate the partial derivative. Therefore, we consider replacing the quaternion with three independent dimensional elements.

Because the error quaternion is a unit quaternion, it can be expressed as

$$\boldsymbol{q_{r_{err}}} = \cos\phi + \vec{u}\sin\phi, \tag{6}$$

where $\vec{u}$ represents the direction of the rotation axis towards the target orientation from the perspective of the world coordinate system, which is a complex unit vector. ϕ represents the rotation angle around $\vec{u}$. If the real part of $\boldsymbol{q_{r_{err}}}$ is defined as q_w and the imaginary part as $\vec{q_v}$, they are defined as follows:

$$\begin{aligned} \vec{u} &= \frac{\vec{q_v}}{||\vec{q_v}||} \\ \phi &= \arctan(||\vec{q_v}||, q_w), \end{aligned} \tag{7}$$

The right side of equation (6) can be expressed as follows using exponential notation [22]:

$$\cos\phi + \vec{u}\sin\phi = e^{\vec{u}\phi}.$$

From this, it can be inferred that taking the logarithm of $\boldsymbol{q_{r_{err}}}$ results in

$$\log(\boldsymbol{q_{r_{err}}}) = \log(e^{\vec{u}\phi}) = \vec{u}\phi. \tag{8}$$

Given that the dimension of $\vec{u}$ is three, it can be inferred that by taking the logarithm of $\boldsymbol{q_{r_{err}}}$, we can obtain a set of three linearly independent equations. The analytical

gradient can be obtained by calculating it in this logarithmic quaternion space and then returning to it the quaternion space by taking its exponent. The mapping from logarithmic to exponential space was obtained using the following formula [27]:

$$\begin{aligned}\frac{\partial \delta\boldsymbol{\theta}}{\partial \delta\phi} \simeq & I+\frac{1}{2}[\vec{u}\phi]_{\times} \\ & +\left(\frac{1}{||\phi||^2}-\frac{1+\cos||\phi||}{2||\phi||\sin||\phi||}\right)[\vec{u}\phi]_{\times}^2,\end{aligned} \tag{9}$$

where, $\delta\boldsymbol{\theta}$ represents the analytical gradient in the quaternion space, and $[]_{\times}$ denotes the skew-symmetric matrix.

The analytical gradient in the quaternion space can be obtained by multiplying Equation (8), which represents the analytical gradient in logarithmic space, by Equation (9). noting that the cross product of the same vectors becomes zero, we obtain

$$\begin{aligned}\delta\boldsymbol{\theta} =& \frac{\partial \delta\boldsymbol{\theta}}{\partial \delta\phi}\log(\boldsymbol{q_{r_{err}}}) \\ =& \vec{u}\phi+\frac{1}{2}[\vec{u}\phi]_{\times}\vec{u}\phi \\ & +\left(\frac{1}{||\phi||^2}-\frac{1+\cos||\phi||}{2||\phi||\sin||\phi||}\right)[\vec{u}\phi]_{\times}^2\vec{u}\phi \\ =& \vec{u}\phi.\end{aligned}$$

As the orientation error decreases, $\vec{q}_v$ converges to zero, but $\vec{u}\phi$ diverges as $\vec{q}_v$ decreases according to its definition. Therefore, we use Taylor's expansion to approximate as follows:

$$\begin{aligned}\delta\boldsymbol{\theta} = \vec{u}\phi &= \frac{\vec{q}_v}{||\vec{q}_v||}\arctan(||\vec{q}_v||, q_w) \\ &\simeq \frac{\vec{q}_v}{||\vec{q}_v||}\left(\frac{||\vec{q}_v||}{q_w}-\frac{||\vec{q}_v||^3}{3{q_w}^3}\right) \\ &= \frac{\vec{q}_v}{q_w}\left(1-\frac{||\vec{q}_v||^2}{3{q_w}^2}\right) \\ &= Q_w\vec{q}_v.\end{aligned} \tag{10}$$

Here $Q_w = \left(\frac{1}{q_w}-\frac{1-{q_w}^2}{3{q_w}^3}\right)$.

The sensitivity of the value Q_w with respect to q_w is shown in Fig. 6:

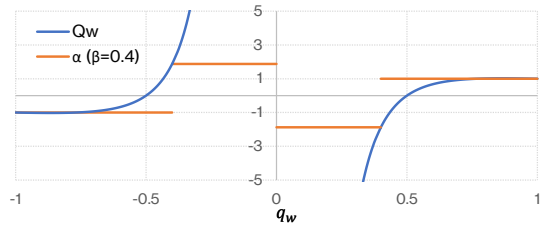


Fig. 6: Sensitivity of analytical gradient

From Fig. 6, we can identify two issues related to the analytical gradient.

- When q_w is small (i.e., the orientation error is large), the analytical gradient tends to diverge toward infinity.
- Because the analytical gradient becomes zero at $q_w = 0.5$, it is prone to getting trapped in a local optimum.

Therefore, in this study, to circumvent these issues, $\delta\boldsymbol{\theta}$ is approximated as follows.

$$\delta\boldsymbol{\theta} = \alpha\vec{q}_v\mathrm{sgn}(q_w). \tag{11}$$

where

$$\alpha = \begin{cases}\frac{1}{\beta}\left(1-\frac{1-\beta^2}{3\beta^2}\right) & : ||q_w|| < \beta \\ 1 & : other\end{cases}.$$

Within the range where $||q_w|| < \beta$, the formula for α equals the formula for Q_w, where q_w is replaced by the threshold β. The value with α respect to q_w is shown in Fig. 6 The threshold β should be less than 0.5 to avoid the an analytical gradient of zero. However, setting an overly low value for β can result in extermely large analytical gradients in regions with large initial pose errors. Therefore, β should not be set to an extremely small value. When $\mathrm{sgn}(q_w) = 0$, $\delta\boldsymbol{\theta}$ becomes zero. This occurs when the target and current postures are orthogonal. An explicit method for avoiding this zero point is to set $\mathrm{sgn}(q_w)$ to 1 when $q_w = 0$.

Next, we consider the quaternion of the position. The update amount of the angle $\delta\boldsymbol{\theta_t}$ based on the position error is the dot product of the fundamental Jacobian matrix $\boldsymbol{J_t}$, which represents the relationship between the position and each joint angle, and the error vector $\vec{q}_{t_{err}}$.

$$\delta\boldsymbol{\theta_t} = \boldsymbol{J_t}\cdot\vec{q}_{t_{err}}.$$

$\boldsymbol{J_t}$ is an $N\times 3$ matrix for an arm with N degrees of rotational freedom, and is defined as follows:

$$\begin{aligned}\boldsymbol{J_t} &= \begin{bmatrix}\vec{t}_0\times\vec{r}_0 & \vec{t}_1\times\vec{r}_1 & \dots & \vec{t}_{N-1}\times\vec{r}_{N-1}\end{bmatrix}^T \\ \vec{t}_n &= \vec{q}_{t_{ref}} - {}^0\vec{q}_{t_n},\end{aligned}$$

where, ${}^0\vec{q}_{t_n}$ represents the vector of the imaginary part of ${}^0\boldsymbol{q}_{\boldsymbol{t}n}$. In addition, when $n=0$ because ${}^0\vec{q}_{t_0}$ does not provide a coordinated translation operation, $\vec{t}_0 = \vec{q}_{t_{ref}}$ holds. For a prismatic joint, the corresponding component J_t represents direction $\vec{r}_n$ of the joint.

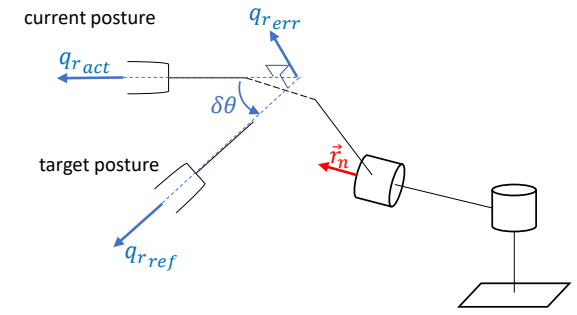
Note that ${}^0\boldsymbol{q}_{\boldsymbol{r}n}$ and ${}^0\vec{q}_{t_n}$, which are needed to calculate $\vec{r}_n$ and $\vec{t}_n$, are determined during the forward kinematics calculations to determine the current position and orientation; thus, $\boldsymbol{J_r}$ and $\boldsymbol{J_t}$ can be computed at a low computational cost.

Because the quaternion of the position does not have a scalar part, we perform the operation as a vector with an imaginary part. $\vec{q}_{t_{ref}}$ is the imaginary part of the target position quaternion, $\vec{q}_{t_{act}}$ is the imaginary part of the current position quaternion, and error vector $\vec{q}_{t_{err}}$ is

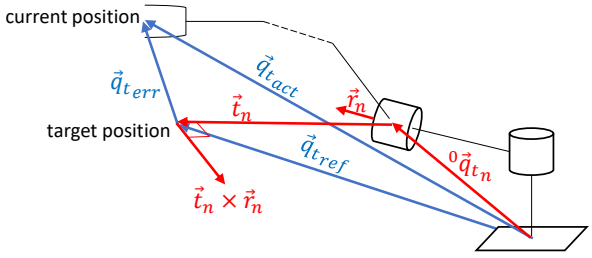
$$\vec{q}_{t_{err}} = \vec{q}_{t_{act}} - \vec{q}_{t_{ref}}.$$

Fig.7 schematically shows the quaternions and vectors used for the inverse kinematics of the robot joints.

The update amount of each joint angle is the sum of $\delta\boldsymbol{\theta_r}$ and $\delta\boldsymbol{\theta_t}$, and by using a weighting matrix, it is also possible to adjust which of the position and orientation to prioritize for each joint.



(a) Relationship with posture



(b) Positional relationship

Fig. 7: Relationship between error from the target and quaternions and vectors. The larger the dot product of $\boldsymbol{q_{r_{err}}}$, which represents the error in orientation, and the rotation axis $\vec{r}_n$, the higher the contribution of the joint to the target orientation, and the greater the update amount of the angle. Moreover, the larger the dot product of the position error $\vec{q}_{t_{err}}$ and $\vec{t}_n \times \vec{r}_n$, the higher the contribution of the joint to the target position, and the greater the update amount of the angle. Since quaternion is a four-dimensional vector, the relationship of orientation is an image.

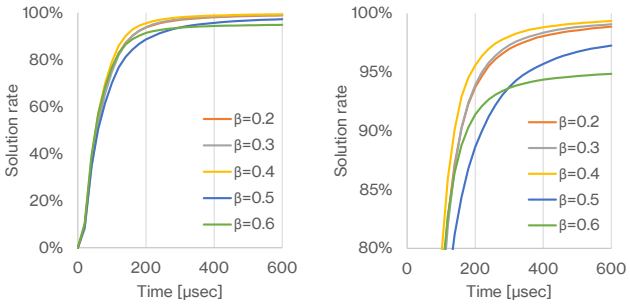
IV. Result and Discussion

To confirm the effectiveness of the proposed method, we measured the calculation time for the inverse kinematics of the robot arm model created in the previous section. In the evaluation, we compared Equation (10) to verify the effect of simplification using Equation (11).

First, we analyze the sensitivity of β in Equation (11) and determine the value of β that yields a highly efficient solution. Next, using the β determined, we conduct a comparative study pertaining to the effects of simplification.

An Intel ®Core i7 2.6GHz was used for the evaluation. The measurements were conducted by generating random target postures within the range of motion of each joint and performing 50,000 calculations.

The sensitivity of β was assessed for values from 0.2 to 0.6 in 0.1 increments. Fig. 8 depicts the solution rate vs. the computation time. At $\beta > 0.5$, the solution rate plateaued at a low value because the original analytical gradient and sign reversal prevented full convergence for $q_w < 0.5$. Although the solution rate increased slightly at $\beta = 0.5$, a longer convergence time was required owing to the necessity for repetitive restarts until $||q_w|| > 0.5$ is reached, which is attributable to the analytical gradient being zero at $\alpha = 0$ within $q_w < 0.5$. When β is overly small, a longer time is required to reach the solution, which is attributable to the large analytical gradient requiring a restart when the initial posture error is large, thus resulting in solution divergence. $\beta = 0.4$ was shown to yielded the best result.

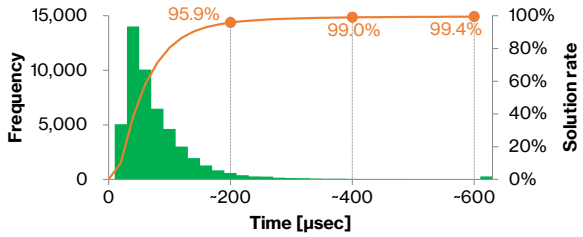


(a) **(b)**

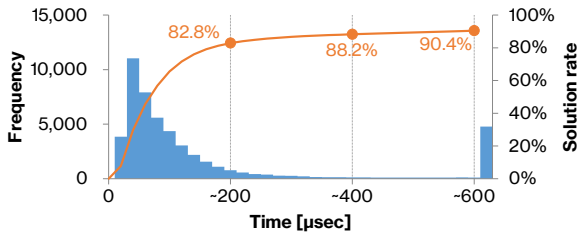
Fig. 8: Sensitivity of β with respect to calculation time. (b) shows enlarged view of more than 80% of (a)

Next, we compare the conventional method using Equation (10) and the proposed method using Equation (11) based on $\beta = 0.4$.

The distribution of the inverse kinematics calculation speeds of the proposed and conventional methods is shown in Fig.9, and the calculation time per loop is listed in Table II. The average number of restarts required in the case of divergence or falling into a local solution was 5.7 times for the proposed method and 7.2 times for the conventional method, which is a 20% reduction compared with the conventional rate.



(a) Proposal Approach ($\beta = 0.4$)



(b) Conventional Approach

Fig. 9: Calculation time for inverse kinematics

As shown in Fig. 9, the proposed method increases the probability of finding a solution simultaneously. According to Table II, there is little difference in the computational speed per loop, suggesting that the improvement in the solution speed is due to the rate of numerical convergence. In

TABLE II: Calculation time of inverse kinematics in one loop

Approach	Mean time [μsec]	Standard deviation [μsec]
Proposal : equation (11)	0.180	0.037
Conventional : equation (10)	0.185	0.036

addition, whereas the conventional method tends to saturate at a solution rate of approximately 90%, the proposed method achieves a solution rate of over 99%, confirming its high robustness.

V. Conclusion

In this study, we propose a method for modeling the coordinate transformation representation using the dual quaternion of a robot by replacing the parallel-link mechanism with an equivalent serial link mechanism. The proposed method was applied to a parallel link mechanism with two degrees of freedom in rotation to achieve a highly efficient coordinate transformation without the use of trigonometric functions. Furthermore, by applying the improved analytical gradient to the inverse kinematics calculation for a composite arm with serial and parallel link mechanisms, we demonstrated that more stable solutions can be obtained than with conventional methods.

Many types of parallel link mechanisms exist, and the generalization of the modeling method proposed in this study is a subject for future research. In addition, because we used finite differences for the inverse kinematic calculation, it was necessary to repeatedly perform the forward kinematic calculations of the robot. We believe that optimization approaches [28] and [29] for obtaining the analytic gradient of the objective function, which can avoid such repeated forward kinematic calculations, would be effective for further improving the computational efficiency of the proposed method.

References

[1] W. R. Hamilton, *Elements of quaternions*. Longmans, Green, & Company, 1866.

[2] X. Wang and H. Zhu, "On the comparisons of unit dual quaternion and homogeneous transformation matrix," *Advances in Applied Clifford Algebras*, vol. 24, pp. 213–229, 2014.

[3] N. A. Aspragathos and J. K. Dimitros, "A comparative study of three methods for robot kinematics," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 28, no. 2, pp. 135–145, 1998.

[4] J. Funda, R. H. Taylor, and R. P. Paul, "On homogeneous transforms, quaternions, and computational efficiency," *IEEE transactions on Robotics and Automation*, vol. 6, no. 3, pp. 382–388, 1990.

[5] A. T. Yang and F. Freudenstein, "Application of dual-number quaternion algebra to the analysis of spatial mechanisms," *Journal of Applied Mechanics*, vol. 31, no. 2, pp. 300–308, 1964.

[6] H.-L. Pham, V. Perdereau, B. V. Adorno, and P. Fraisse, "Position and orientation control of robot manipulators using dual quaternion feedback," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 658–663, IEEE, 2010.

[7] M. d. P. A. Fonseca, B. V. Adorno, and P. Fraisse, "Coupled task-space admittance controller using dual quaternion logarithmic mapping," *IEEE Robotics and Automation Letters*, vol. 5, no. 4, pp. 6057–6064, 2020.

[8] R. N. Colomba, R. Laha, L. F. Figueredo, and S. Haddadin, "Adaptive admittance control for cooperative manipulation using dual quaternion representation and logarithmic mapping," in *2022 IEEE 61st Conference on Decision and Control (CDC)*, pp. 107–144, IEEE, 2022.

[9] H. Abaunza, R. Chandra, E. Özgür, J. A. C. Ramón, and Y. Mezouar, "Kinematic screws and dual quaternion based motion controllers," *Control Engineering Practice*, vol. 128, p. 105325, 2022.

[10] L. Figueredo, B. V. Adorno, J. Y. Ishihara, and G. A. Borges, "Robust kinematic control of manipulator robots using dual quaternion representation," in *2013 IEEE International Conference on Robotics and Automation*, pp. 1949–1955, IEEE, 2013.

[11] B. V. Adorno, P. Fraisse, and S. Druon, "Dual position control strategies using the cooperative dual task-space framework," in *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pp. 3955–3960, IEEE, 2010.

[12] F. F. Afonso Silva, J. José Quiroz-Omaña, and B. Vilhena Adorno, "Dynamics of mobile manipulators using dual quaternion algebra," *Journal of Mechanisms and Robotics*, vol. 14, no. 6, p. 061005, 2022.

[13] J. Cheng, J. Kim, Z. Jiang, and W. Che, "Dual quaternion-based graphical slam," *Robotics and Autonomous Systems*, vol. 77, pp. 15–24, 2016.

[14] X. Wang, C. Yu, and Z. Lin, "A dual quaternion solution to attitude and position control for rigid-body coordination," *IEEE Transactions on Robotics*, vol. 28, no. 5, pp. 1162–1170, 2012.

[15] H. J. Savino, L. C. Pimenta, J. A. Shah, and B. V. Adorno, "Pose consensus based on dual quaternion algebra with application to decentralized formation control of mobile manipulators," *Journal of the Franklin Institute*, vol. 357, no. 1, pp. 142–178, 2020.

[16] B. V. Adorno, *Two-arm manipulation: From manipulators to enhanced human-robot collaboration*. PhD thesis, Université Montpellier II-Sciences et Techniques du Languedoc, 2011.

[17] X. Yang, H. Wu, Y. Li, and B. Chen, "A dual quaternion solution to the forward kinematics of a class of six-dof parallel robots with full or reductant actuation," *Mechanism and Machine Theory*, vol. 107, pp. 27–36, 2017.

[18] V. Noppeney, T. Boaventura, and A. Siqueira, "Task-space impedance control of a parallel delta robot using dual quaternions and a neural network," *Journal of the Brazilian Society of Mechanical Sciences and Engineering*, vol. 43, no. 9, p. 440, 2021.

[19] D. Chevallier, "Lie algebras, modules, dual quaternions and algebraic methods in kinematics," *Mechanism and Machine Theory*, vol. 26, no. 6, pp. 613–627, 1991.

[20] M. Gouasmi, "Robot kinematics, using dual quaternions," *IAES International Journal of Robotics and Automation*, vol. 1, no. 1, p. 13, 2012.

[21] D.-P. Han, Q. Wei, and Z.-X. Li, "Kinematic control of free rigid bodies using dual quaternions," *International Journal of Automation and Computing*, vol. 5, pp. 319–324, 2008.

[22] X. Wang, D. Han, C. Yu, and Z. Zheng, "The geometric structure of unit dual quaternion with application in kinematic control," *Journal of Mathematical Analysis and Applications*, vol. 389, no. 2, pp. 1352–1364, 2012.

[23] E. Özgür and Y. Mezouar, "Kinematic modeling and control of a robot arm using unit dual quaternions," *Robotics and Autonomous Systems*, vol. 77, pp. 66–73, 2016.

[24] N. T. Dantam, "Robust and efficient forward, differential, and inverse kinematics using dual quaternions," *The International Journal of Robotics Research*, vol. 40, no. 10-11, pp. 1087–1105, 2021.

[25] K. Nishii, A. Hatano, and Y. Okumatsu, "Ultra-low inertia 6-dof manipulator arm for touching the world," in *2023 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 432–438, IEEE, 2023.

[26] J.-P. Merlet, *Parallel robots*, vol. 128. Springer Science & Business Media, 2006.

[27] G. S. Chirikjian, *Stochastic models, information theory, and Lie groups, volume 2: Analytic methods and modern applications*, vol. 2. Springer Science & Business Media, 2011.

[28] P. Beeson and B. Ames, "Trac-ik: An open-source library for improved solving of generic inverse kinematics," in *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*, pp. 928–935, IEEE, 2015.

[29] S. Kumar, N. Sukavanam, and R. Balasubramanian, "An optimization approach to solve the inverse kinematics of redundant manipulator," *International Journal of Information and System Sciences (Institute for Scientific Computing and Information)*, vol. 6, no. 4, pp. 414–423, 2010.