Kinematic Synthesis of a Serial Robotic Manipulator by Using Generalized Differential Inverse Kinematics

Shouhei Shirafuji, *Member, IEEE*, and Jun Ota, *Member, IEEE*

***Abstract*—In this paper, we propose an optimization method to determine the design parameters comprising joint displacement parameters in a serial manipulator having small degrees of freedom realizing an approximated target trajectory. Using generalized differential kinematics, we can achieve efficient optimization without solving the inner-loop optimization required to obtain the reachable points.**

***Index Terms*—Kinematic synthesis, generalized differential inverse kinematics, optimization of design, serial manipulator.**

I. Introduction

Articulated six-axis or seven-axis robotic arms are often used in manufacturing when complex three-dimensional (3-D) motions are required (e.g., automatic welding of a car body). A manipulator with six degrees of freedom (DoFs) is typically selected because it can move its end-effector to the desired position and orientation. However, a seven-axis robotic arm is preferred when there is a risk of robot self-collisions or collisions of robots with environment. At the same time, majority of the robotic arms in manufacturing lines move in specified trajectories and do not require to engage all the DoFs. For example, the manipulator can use a very small number of DoFs if a task requires the end-effector of the robot to loosely follow a planned path (e.g. to avoid collisions). A robotic manipulator with the minimum DoFs required to accomplish a specific task reduces the energy used to operate the manipulator and the cost of the manipulator itself. The demand for such task-specific designs of manipulators has been increasing in recent years because of the advances in manufacturing technology (including 3-D printing). However, it is difficult to manually design a robotic arm with complex joint displacements for a specific task. Joint displacement optimization is one methodology to solve this problem and realize flexible and task-specific manipulator designs.

On the basis of various indices, optimal designs and kinematic syntheses of serial and parallel manipulators have been extensively discussed, especially in the 1980s and 1990s, from the viewpoints of reachable and dexterous workspaces [1], [2] (e.g. of a serial chain with three revolute joints [3], six revolute joints [4]–[7], or other combinations of joints [8]). An optimal design of 3-DoF serial and parallel manipulators based on the condition number of the Jacobian matrix was proposed in [9] and [11], which has been considered in several other works as well [10]–[12]. Other criteria, including the avoidance of a singular configuration [13], [14], manipulability [15], and some requirements to accomplish the given tasks [16], have been discussed from the viewpoint of an optimal robotic manipulator design.

Manuscript received December 2, 2018; accepted March 9, 2019. Date of publication April 26, 2019; date of current version August 1, 2019. This work was supported by the JSPS KAKENHI under Grant 16H06682 and Grant 18K13724. This paper was recommended for publication by Associate Editor Q.-C. Pham and Editor I.-M. Chen upon evaluation of the reviewers' comments. *(Corresponding author: Shouhei Shirafuji.)*

The authors are with the Research into Artifacts Center, Center for Engineering, School of Engineering, The University of Tokyo, Tokyo 113-8656, Japan (e-mail: shirafuji@race.t.u-tokyo.ac.jp; ota@race.t.u-tokyo.ac.jp).

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org, provided by the authors.

Digital Object Identifier 10.1109/TRO.2019.2907810

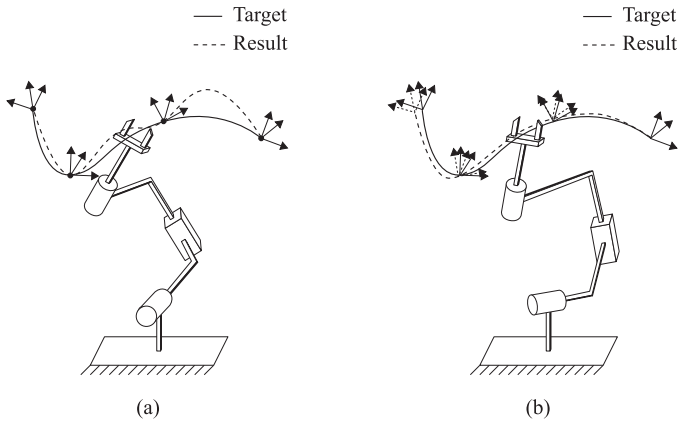

Fig. 1. Conceptual illustrations of two types of motion synthesis for a given trajectory. (a) Kinematic synthesis by selecting several positions on a given trajectory. (b) Kinematic synthesis by approximating a given trajectory.

Researchers have discussed the kinematic synthesis of simple modular robots for a given task based on several criteria, such as the workspace volume, positioning accuracy, mechanical simplicity, and solvability of inverse kinematics [17], [18]. In addition, a computation method has been proposed for the kinematic synthesis and design of a manipulator based on its dynamics [19], [20]. By considering the dynamics of the system, Ha *et al.* [21] proposed a flexible computational approach to optimize the design of the open- and closed-loop robots with the motion trajectory itself by considering the trajectory as the design parameter and using the implicit function theorem.

The studies on the problem addressed in this paper focused mainly on kinematic synthesis for realizing a serial chain manipulator that can help achieve given end-effector configurations. The algebraic equations to realize the given configurations have been formulated for many types of serial chains, including serial chains with two revolute joints [22], two cylindrical joints [23], three revolute joints [24], [25], and more complicated combinations of joints [26]–[29]; some of them were solved algebraically, whereas others were solved numerically [30]–[32]. A generalized method was proposed in [33] and [34] to determine the number of positions of the end-effector to specify the design parameters of a given serial chain and to derive equations to be solved using Clifford algebra. The authors have numerically solved the derived equations.

In various cases, where a continuous trajectory of the end-effector is provided, there is no kinematic-synthesis-based solution for a serial chain manipulator with five or fewer joints that can follow the specified trajectory. Kinematic synthesis by selecting as many positions from the end-effector trajectory as required to specify the structural parameters was realized in [32] and [34] [see Fig. 1(a)]. By contrast, in this paper, we propose a methodology to derive a serial chain having fewer than five joints to realize an approximate end-effector trajectory that is as close as possible to a given trajectory without choosing any specific positions [see Fig. 1(b)]. This approach is useful when certain important

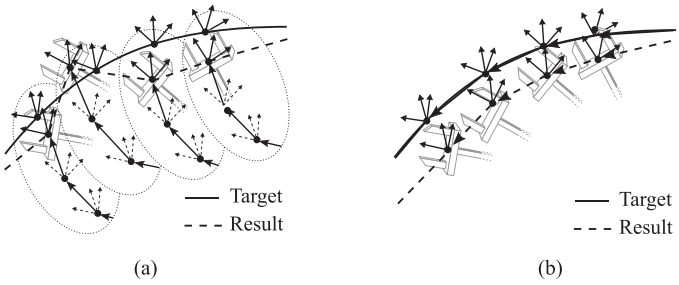

Fig. 2. Conceptual illustrations of the two approaches to obtain the trajectory of the end-effector realized by having the given joint displacements as close as possible to a given trajectory. (a) The approach that uses iterative calculation to obtain the closest position and orientation over the trajectory. (b) The approach that uses differential inverse kinematics to follow a given trajectory as close as possible.

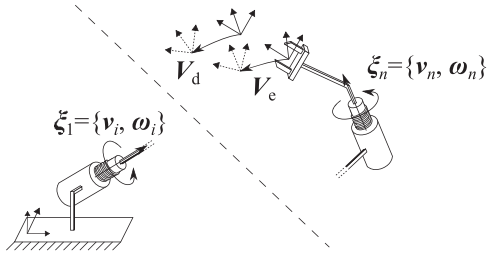

Fig. 3. Target and actual derivation of the end-effector realized by the manipulator specified by the set of twists.

positions are difficult to choose from the target trajectory or when a smooth trajectory is required to follow the target trajectory.

A straightforward methodology to solve this task is the optimization of the joint dispositions by solving the inverse kinematics problem for a given joint disposition over a given end-effector trajectory and evaluating the errors between the target and the result. The problem with such an approach is that the computation is expensive. Recently, the inverse kinematics of a wide variety of serial manipulators have been solved quickly using an analytical method [35]. However, for a manipulator with fewer than six DoFs, inverse kinematics may not have any exact solutions for a given end-effector's position and orientation in many cases. No general analytical method has been proposed to obtain the joint configurations that realize small errors in the end-effector's position and orientation with respect to the target; majority of the methods require at least numerical calculations, such as the iterative procedure to reach the closest position and orientation to those of the end-effector for the given joint displacements. Our objective is to derive joint displacements to follow the target trajectory of the end-effector's position and orientation as close as possible, even in the case of small DoFs. Therefore, an iterative procedure is required for the evaluation of joint displacements in kinematic synthesis and for solving inverse kinematics over a given trajectory, as depicted in Fig. 2(a). The duplicated optimization process for the inverse kinematics and structural parameters is computationally expensive.

Therefore, in this paper, we calculate the trajectory realized by employing the joint displacements obtained in an optimization step by using differential inverse kinematics. Our optimization method evaluates the joint displacements by comparing the calculated and target trajectories. Differential inverse kinematics derives the velocities of joints from their relation represented by the Jacobian to realize the targeted velocity of the end-effector. Its advantage is the pseudo-inverse of the Jacobian that provides a unique solution of the joint velocities, making the velocity of the end-effector to be close to the target velocity in the least squares sense. The trajectory realized by the joint displacements to follow the target trajectory can be obtained by calculating the differential inverse over the given trajectory, as depicted in Fig. 2(b). The joint displacements can be evaluated by checking the error between the calculated and target trajectories. This methodology makes it possible to avoid the dual optimization of the inverse kinematics and joint displacements. Furthermore, we use the generalized differential inverse [36] to evaluate the errors in the position and orientation equivalently.

The rest of this paper is organized as follows. In Section II, we introduce the proposed methodology for kinematic synthesis. The section also describes the representation of the trajectory and the structure of the manipulator using the twist and the methodology to calculate the error for a given trajectory and structural parameters. In addition, Section II outlines the procedure for optimizing the structural parameters based on this error. In Section III, the proposed method is applied to examples of the kinematic synthesis problem.

II. Methodology

A. Definition of Target Trajectory and the Given Joint Displacement

Let $\boldsymbol{p}_{\mathrm{d}}(s) \in \mathbb{R}^3$ be a given trajectory of the position of an end-effector of a manipulator, and $\boldsymbol{R}_{\mathrm{d}}(s) \in SO(3)$ be a given trajectory of the orientation of the end-effector relative to the spatial frame. The parameter $s \in \mathbb{R}$ is used to represent these trajectories. The derivative of the motion of the rigid body fixed at the end of the manipulator relative to the spatial frame is expressed as follows:

$$\widehat{\boldsymbol{V}}_{\mathrm{d}} = \begin{bmatrix} \boldsymbol{R}_{\mathrm{d}}' \boldsymbol{R}_{\mathrm{d}}^T & -\boldsymbol{R}_{\mathrm{d}}' \boldsymbol{R}_{\mathrm{d}}^T \boldsymbol{p}_{\mathrm{d}} + \boldsymbol{p}_{\mathrm{d}}' \\ 0 & 0 \end{bmatrix}. \tag{1}$$

In this paper, p' is Lagrange's notation to represent the derivative of a variable p with respect to the parameter s. The twist coordinate of the derivative shown in Fig. 3 is expressed as follows:

$$\boldsymbol{V}_{\mathrm{d}} = \begin{bmatrix} \boldsymbol{R}_{\mathrm{d}}^T \boldsymbol{p}_{\mathrm{d}}' \\ (\boldsymbol{R}_{\mathrm{d}}^T \boldsymbol{R}_{\mathrm{d}}')^{\vee} \end{bmatrix} \tag{2}$$

where $\vee$ represents the transformation that projects the rotational matrix onto the corresponding rotational axis [37].

The joint displacements also are represented by the twist coordinates in this method. Let the serial manipulator have n joints. When the norms of the joint twists do not affect the resultant design of the joints, the ith joint can be represented by twist $\boldsymbol{\xi}_i \in \mathbb{R}P^5$. Here, $\mathbb{R}P^5$ refers to the real projective space of dimension five. The kinematic structure of the manipulator is described by the set of all joint twists as $\boldsymbol{\Xi} = \{\boldsymbol{\xi}_1, \boldsymbol{\xi}_2, \ldots, \boldsymbol{\xi}_n\}$.

As explained in the previous section, differential inverse kinematics is used to derive the structural parameters for following a given trajectory. When $\boldsymbol{\Xi}$ is given, the derivative of the end position of the manipulator $\boldsymbol{V}_{\mathrm{e}}$ is given by

$$\boldsymbol{V}_{\mathrm{e}} = \boldsymbol{J}_{\boldsymbol{\Xi}}(\boldsymbol{\theta}, \boldsymbol{\Xi})\boldsymbol{\theta}' \tag{3}$$

where $\boldsymbol{\theta} = [\theta_1, \theta_2, \ldots, \theta_n] \in \mathbb{R}^n$ is the joint configuration of the manipulator relative to the referential position decided appropriately, and $\boldsymbol{J}_{\boldsymbol{\Xi}} \in \mathbb{R}^{6\times n}$ is a spatial manipulator Jacobian defined by using the joint twists as follows:

$$\boldsymbol{J}_{\boldsymbol{\Xi}}(\boldsymbol{\theta}, \boldsymbol{\Xi}) = \left[\boldsymbol{\xi}_1 \ \mathbf{Ad}_{\left(e^{\widehat{\boldsymbol{\xi}}_1 \theta_1}\right)} \boldsymbol{\xi}_2 \ \cdots \ \mathbf{Ad}_{\left(e^{\widehat{\boldsymbol{\xi}}_1 \theta_1} \ldots e^{\widehat{\boldsymbol{\xi}}_{i-1} \theta_{i-1}}\right)} \boldsymbol{\xi}_i \right] \tag{4}$$

where $\mathbf{Ad}_g \in \mathbb{R}^{6\times 6}$ is an adjoint matrix that transforms a twist from one coordinate to another coordinate given by $g \in SE(3)$, and $e^{\widehat{\boldsymbol{\xi}}_1 \theta_1} \cdots e^{\widehat{\boldsymbol{\xi}}_{i-1} \theta_{i-1}}$ is the product of the exponential formulations of the manipulator kinematics [32], [37]. Therefore, $\boldsymbol{\Xi}$ represents the joint displacements when they are in referential configuration, that is, $\boldsymbol{\theta} = \mathbf{0}$.

Fig. 4. Types of joints: (a) screw joint, (b) revolute joint, and (c) prismatic joint.

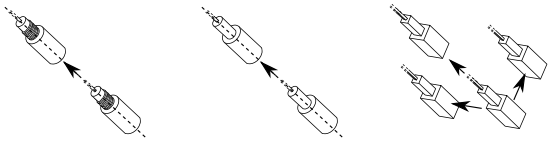

Fig. 5. Shifts of joints with which the end-effector motion is invariant: (a) screw joint, (b) revolute joint, and (c) prismatic joint.

The joints represented by the twists generate screw motions, which imply that the motion of the rigid body is defined by proportionate transition and rotation, as shown in Fig. 4(a). This type of joint may be realized by a ball screw with a given pitch, but a prismatic joint or a revolute joint are used more widely as the mechanical components of the manipulator joints. For designing a manipulator composed of revolute [see Fig. 4(b)] and prismatic [see Fig. 4(c)] joints, we need to constrain the twist in the optimization as follows:

$$\boldsymbol{v}_i^T \boldsymbol{\omega}_i = 0 \qquad \text{when the } i \text{ th joint is revolute} \tag{5}$$

$$\boldsymbol{\omega}_i = \boldsymbol{0} \qquad \text{when the } i \text{ th joint is prismatic} \tag{6}$$

where $\boldsymbol{v}_i$ and $\boldsymbol{\omega}_i$ are the transitional and rotational components, respectively, and $\boldsymbol{\xi}_i = \{\boldsymbol{v}_i, \boldsymbol{\omega}_i\}$. Other types of joints including the cylindrical, plane, and spherical joints, can be regarded as the combinations of the revolute and prismatic joints with a few additional constraints [32], but these joints are not relevant to the scope of this paper.

The twist representation of the joint describes the motion of the end-effector when this joint has instantaneous movement. In these types of joints, joint displacements can shift to some specific direction without having any effect on the end-effector motion caused by the shifted joint, as shown in Fig. 5. The end-effector motion caused by a screw joint is invariant with the shifting of the joint displacement in the direction of its joint axis, as shown Fig. 5(a). In the case of the revolute joint, the invariant in the end-effector motion also shifts in the direction of the joint axis, as shown Fig. 5(b). The motion caused by a prismatic joint is invariant to shifting in every direction, as shown Fig. 5(c). Therefore, the joint represented in this twist does not specify the displacements in these directions. Although the decision of joint displacements in these invariant directions is important to avoid self-collision and to avoid collisions with the environment and the large-scale manipulator; we do not deal with the joint displacements in the invariant direction because they do not change the resultant motion of the end-effector.

B. Derivation of Trajectory Tracking Error

In this paper, our aim is to find a suitable $\boldsymbol{\Xi}$ to generate a trajectory for the end-effector of the manipulator, which is as close as possible to the desired trajectory. In other words, the structural parameters $\boldsymbol{\Xi}$ are optimized by calculating the errors between the target derivative $\boldsymbol{V}_{\mathrm{d}}$ and the realized derivative $\boldsymbol{V}_{\mathrm{e}}$ of the manipulator by using the structural parameters $\boldsymbol{\Xi}$. Therefore, we need to calculate the derivatives of joint angles that realize the desired trajectory of the end position of the manipulator. The inverse of the Jacobian is used to derive the instantaneous derivative of the joint angles to achieve instantaneous motion that is close to the derivative of the target trajectory. In many cases where the relationship between the joint configuration and the configuration of the end-effector is not bijective, the Moore–Penrose pseudo-inverse of the Jacobian is used to solve the inverse problem of the derivatives of the joints and the end-effector. However, Doty *et al.* [36] pointed out that the Moore–Penrose pseudo-inverse of the Jacobian, which represents the relationship between the twists, is inappropriate in a few cases because of the differences in the units between the rotation and the translation, and they proposed a generalized inverse of the Jacobian, which is expressed as follows:

$$\boldsymbol{J}_{\boldsymbol{\Xi}}^{\#} = \left[\boldsymbol{J}_{\boldsymbol{\Xi}}^T \boldsymbol{M}_{\mathrm{v}} \boldsymbol{J}_{\boldsymbol{\Xi}}\right]^{-1} \boldsymbol{J}_{\boldsymbol{\Xi}}^T \boldsymbol{M}_{\mathrm{v}} \tag{7}$$

when $n < 6$. Here, $\boldsymbol{M}_{\mathrm{v}}$ is a weighting matrix. We can choose from many types of weighting matrices. Researchers have used a kinetic energy metric in [36] and [38]. They proposed this metric based on the fact that the kinetic energy of a rigid body is a physical quantity obtained by its rotational and translational motions, and this metric does not depend on the referential frame. As a result, we can obtain the generalized inverse, which is physically invariant for the differences among its units and its referential frame. The kinetic energy metric for the end-effector with respect to the body frame fixed on its center of mass is given as follows:

$$\boldsymbol{M}_{\mathrm{v}}^{\mathrm{b}} = \begin{bmatrix} m\boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{0} & \mathcal{I} \end{bmatrix} \tag{8}$$

where m is the mass and $\mathcal{I}$ the moment of inertia tensor. $\boldsymbol{M}_{\mathrm{v}}$ is obtained by transforming the body frame into the spatial frame. Therefore, a least-squared, minimum-norm solution of the derivatives of the joint configurations for the target derivative of the end position is given as follows:

$$\boldsymbol{\theta}' = \boldsymbol{J}_{\boldsymbol{\Xi}}^{\#} \boldsymbol{V}_{\mathrm{d}}. \tag{9}$$

Using (9), the end-effector is gradually moved to follow the desired trajectory. This is basically the same as the case of the control method for robotic manipulators, called the pseudo-inverse control [39]. As the result, the resultant trajectory of the end-effector when it was moved using (9) and the sum of the squared differences between the resultant and desired trajectory is defined as follows:

$$e = \int_0^{s^e} \left(\boldsymbol{V}_{\mathrm{d}} - \boldsymbol{J}_{\boldsymbol{\Xi}} \boldsymbol{J}_{\boldsymbol{\Xi}}^{\#} \boldsymbol{V}_{\mathrm{d}}\right)^T \boldsymbol{M}_{\mathrm{v}} \left(\boldsymbol{V}_{\mathrm{d}} - \boldsymbol{J}_{\boldsymbol{\Xi}} \boldsymbol{J}_{\boldsymbol{\Xi}}^{\#} \boldsymbol{V}_{\mathrm{d}}\right) \mathrm{d}s \tag{10}$$

where $\boldsymbol{V}_{\mathrm{d}}(0)$ is the target derivative in the initial configuration and $\boldsymbol{V}_{\mathrm{d}}(s^e)$ the target derivative at the end of the trajectory. $\boldsymbol{M}_{\mathrm{v}}$ is multiplied; therefore, (10) is the accumulation of the instantaneous kinetic energy required to adjust the resultant end-effector's velocity to the target velocity if the parameter s is time. As a result, the variants originating from the differences in the units of translation and rotation and the differences between the referential frames are compensated for the same reason as stated above.

In the following sections, we assume that the initial configuration is the referential configuration, that is $\boldsymbol{\theta}(0) = \boldsymbol{0}$. Therefore, $\boldsymbol{\Xi}$ represents the joint displacements when the manipulator is in the initial configuration in the trajectory. We note that the method optimizes the joint displacements based on the approximated trajectory by differential inverse, and the end-effector on the initial placement gradually drifts from the target in the approximated trajectory. The joint displacements and the generated instantaneous rigid body motions of the end-links of these joints are considered in the proposed optimization process, but the end-effector's exact position and orientation are not specified in this optimization process, even though the target's rigid body motion is defined by the target motion of the end-effector given by (1). This means that we set the target in the optimization as the trajectory of the rigid body motion but not as the point specified on the rigid body. Therefore, we can set the end-effector on the trajectory after the optimization. Although setting the end-effector to coincide with the end-effector on the initial point of the target trajectory is intuitive, as shown in Fig. 6(a), we also set it to coincide with the target end-effector's position and orientation on any point of the trajectory, as shown in

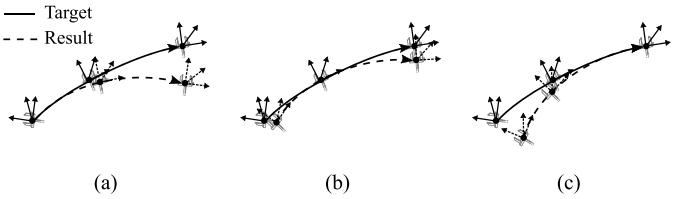

Fig. 6. End-effector's trajectory depending on its initial displacement.

Fig. 6(b) and (c). Nevertheless, that choice does not affect the objective given by (10) because the choice does not change the instantaneous rigid body motion with respect to the spatial frame.

C. Optimization of Structural Parameters

The types of joints and their displacements are calculated by minimizing error e. To do this, we solve the following optimization problem:

$$\underset{\Xi}{\text{minimize}} \quad e. \tag{11}$$

When these twists are represented by the spherical coordinates of the surface of a hemisphere, each twist $\boldsymbol{\xi}_i$ is described by using five variables, $\{\phi_{i1}, \phi_{i2}, \phi_{i3}, \phi_{i4}, \phi_{i5}\}$, where $\phi_{i1}, \phi_{i2}, \phi_{i3}, \phi_{i4} \in [0, \pi]$, and $\phi_{i5} \in [0, \pi)$ are parameters of the angular coordinates. Each twist is represented by using these parameters as follows:

$$\widehat{\boldsymbol{\xi}}_i = \begin{bmatrix} \cos\phi_{i1} \\ \sin\phi_{i1}\cos\phi_{i2} \\ \sin\phi_{i1}\sin\phi_{i2}\cos\phi_{i3} \\ \sin\phi_{i1}\sin\phi_{i2}\sin\phi_{i3}\cos\phi_{i4} \\ \sin\phi_{i1}\sin\phi_{i2}\sin\phi_{i3}\sin\phi_{i4}\cos\phi_{i5} \\ \sin\phi_{i1}\sin\phi_{i2}\sin\phi_{i3}\sin\phi_{i4}\sin\phi_{i5} \end{bmatrix}. \tag{12}$$

As a result, this problem can be regarded as an optimization problem in the bounded $\mathbb{R}^{5\times n}$ space, where n is the number of joints.

The optimization problem in kinematic analysis or synthesis can be solved efficiently if they are described in some quadratic form [40]–[42]. However, (10) is not in a quadratic form because of the integral. Therefore, we chose a derivative-free optimization methodology to solve the problem. Many types of derivative-free optimization methods are available [43]. An example of solving the optimization problem by using a derivative-free method, called multilevel coordinate search [44], is shown in Section III.

When the joint is revolute or prismatic, we can use solve the optimization problem for other parameters. When the ith joint is a revolute joint, the corresponding twist can be represented by using four angular parameters $\{\phi_{i1}, \phi_{i2}, \phi_{i3}, \phi_{i4}\}$, where $\phi_{i1} \in [0, \pi)$ and $\phi_{i2}, \phi_{i3}, \phi_{i4} \in [0, \pi]$. The twist is given by

$$\widehat{\boldsymbol{\xi}}_i = \begin{bmatrix} \cos\phi_{i1}\cos\phi_{i4} \\ \sin\phi_{i1}\cos\phi_{i2}\cos\phi_{i4} \\ \sin\phi_{i1}\sin\phi_{i2}\cos\phi_{i4} \\ -\sin\phi_{i1}\cos\phi_{i3}\sin\phi_{i4} \\ (\cos\phi_{i1}\cos\phi_{i2}\cos\phi_{i3} - \sin\phi_{i2}\sin\phi_{i3})\sin\phi_{i4} \\ (\cos\phi_{i1}\sin\phi_{i2}\cos\phi_{i3} + \cos\phi_{i2}\sin\phi_{i3})\sin\phi_{i4} \end{bmatrix} \tag{13}$$

when the ith joint is prismatic; the corresponding twist can be represented by two angular parameters $\{\phi_{i1}, \phi_{i2}\}$, where $\phi_{i1} \in [0, \pi]$ and $\phi_{i2} \in [0, \pi)$. The twist is given by

$$\widehat{\boldsymbol{\xi}}_i = \begin{bmatrix} \cos\phi_{i1} \\ \sin\phi_{i1}\cos\phi_{i2} \\ \sin\phi_{i1}\sin\phi_{i2} \\ 0 \\ 0 \\ 0 \end{bmatrix}. \tag{14}$$

Therefore, the adequate dispositions of the revolute and prismatic joints that accomplish the given trajectory are determined by solving the optimization problem for these parameters.

III. Examples of Optimization

In this section, we give an example of the optimization of joint disposition. The design target of this example is a robotic manipulator drawing a letter on a surface with a specific number of joints. We show three examples when the robotic manipulator draws 1) the letter "T" on a flat surface without changing the orientation of the end-effector, 2) the letter "O" on a flat surface by rotating the end-effector, and 3) the letter "R" on an egg-shaped object by maintaining the end-effector normal to the surface of the object. These three examples correspond to the following three problems: 1) the problem of a simple planar motion having a trivial solution; 2) the problem of simple planar motion having no trivial solution; and 3) the problem of a complex motion in 3-D space. We will compare the results for these problems under various conditions to discuss the proposed method.

For every case, the target trajectory needs to be discretized to compute the evaluation function (10) and solve the problem. Let the numerical differential of the discretized trajectory with respect to the body frame be $\boldsymbol{V}_{\mathrm{d}}^{\mathrm{b}}(s)$, where $s \in \{0, 1, \ldots, s^{\mathrm{e}}\}$ is a parameter that specifies the trajectory; s^{e} corresponds to the sample number here. In the following examples, we discretized the trajectory so that the weighted norm of the numerical differential will be constant. Weighting compensates for the difference of units between the translational and rotational components in $\boldsymbol{V}_{\mathrm{d}}^{\mathrm{b}}$ and is represented by a twist. We define a weighting matrix as follows:

$$\boldsymbol{N} = \begin{bmatrix} c_v\boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{0} & c_\omega\boldsymbol{I} \end{bmatrix} \tag{15}$$

where c_v and c_ω are the constants for the weighting translational and rotational components, respectively. These constants determine the balance between the translational and rotational components in the discretization. The resultant discretized trajectory is selected such that the following is satisfied:

$$\|\boldsymbol{N}\boldsymbol{V}_{\mathrm{d}}^{\mathrm{b}}\| = u \tag{16}$$

where $u \in \mathbb{R}$ is a given value of the norm. This norm determines the number of discretized points. In this paper, we do not examine the optimal decision of those parameters; although this decision affects the accuracy of the approximated trajectory calculation. High-resolution in the discretization increases the accuracy involved in following the target trajectory, but it makes the computation expensive. In the examples, we decided c_v and c_ω to make the maximum ratio between the translation and the rotation on the target trajectory small. On the basis of the decided c_v and c_ω, we adjusted u by checking the resultant number of the discretized points. Table I lists the parameters for each example including the resultant number of the discreted points on the trajectory.

For calculating the generalized inverse of the Jacobian represented by (7), the kinetic energy metric represented by (8) requires inertia parameters. Although the inertia parameter of the end-effector or the object held by it will be used as the measure in the actual designing procedure, we used the constants c_v and c_ω as the inertia parameters instead of the actual parameters for the sake of simplicity as follows:

$$m = c_v, \qquad \mathcal{I} = c_\omega\boldsymbol{I}. \tag{17}$$

TABLE I
PARAMETERS, RESULTANT DISCRETE-POINT NUMBER, AND RESULTANT COMPUTATION TIME

Example number	c_v	c_ω	u	Number of discretized point	Computation time (seconds)
III-A	1.0	1.0	0.001	168	6.74
III-B	27.5	1.0	0.05	312	14.32
III-C1	10.9	1.0	0.01	756	77.15
III-C2	10.9	1.0	0.01	756	320.36
III-C3	10.9	1.0	0.01	756	101.70

We set $c_v = 10.9$ [kg] and $c_\omega = 1.0$ [kg.m^2] in the following examples. Furthermore, we regard parameter s as time for clarity of the physical meaning of the error function.

We used MATLAB (Mathworks, Natick, MA, USA) for our calculations. We chose the multilevel coordinate search [44] to solve the optimization problem in (11) by using the procedure mentioned in the previous section, and we used the implementation of the multilevel coordinate search on MATLAB [45] created by one of the authors of the work in [44]. Table I also shows the computation time for the examples; these values were obtained by using a MacBook Pro with a 3.3 GHz Intel Core i7 processor and 16 GB RAM.

A. Drawing "T" on a Plane With Two Joints

In our first example, we designed a manipulator with two joints for drawing the letter "T" on a flat surface without changing the end-effector's orientation, as shown in Fig. 7(a). This simple design problem had a trivial solution for following the path with two joints; the two prismatic joints had axes in different directions on the plane of the flat surface. The end-effector of the designed manipulator could completely follow the given trajectory. In this example, we show that our methodology derives this trivial solution.

The twists obtained are as follows:

$$\boldsymbol{\xi}_1 = \begin{bmatrix} -0.5670 \\ 0.8237 \\ 0.0000 \\ 0.0000 \\ 0.0000 \\ 0.0000 \end{bmatrix}, \quad \boldsymbol{\xi}_2 = \begin{bmatrix} 1.0000 \\ 0.0000 \\ 0.0000 \\ 0.0000 \\ 0.0000 \\ 0.0000 \end{bmatrix}. \tag{18}$$

Fig. 7(b) shows the resultant joint displacements in the initial orientation of the manipulator and the trajectory, which is generated when the manipulator is controlled to follow the desired trajectory based on the generalized inverse represented by (9). We can see that $\boldsymbol{\omega}_1 = 0$ and $\boldsymbol{\omega}_2 = 0$ in (18). Therefore, the obtained joints are prismatic joints. Furthermore, we can also see that the direction of the joint axes is in the xy plane. Consequently, the obtained joints are a trivial solution, as we expected. As a result, the end-effector of the manipulator can follow the path correctly, and the sum of squared differences was $e = 8.46 \times 10^{-37}$ [J.s], which was quite a small error.

We note that the relative direction of the joint axes was not uniquely determined in this problem. Furthermore, the joint axes were not orthogonal to each other because there were no restrictions for making them orthogonal in our method. As a result, one solution was chosen in the optimization process.

B. Drawing "O" on a Plane With Two Joints

The second example is the design of the manipulator with two joints for drawing the letter "O" on a flat surface. This is different from the previous example because in this example, while following the path to draw the letter "O," the end-effector gradually rotated about the axis perpendicular to the surface [see Fig. 8(a)]. It is obvious that the solution of this design problem does not completely follow the target trajectory,

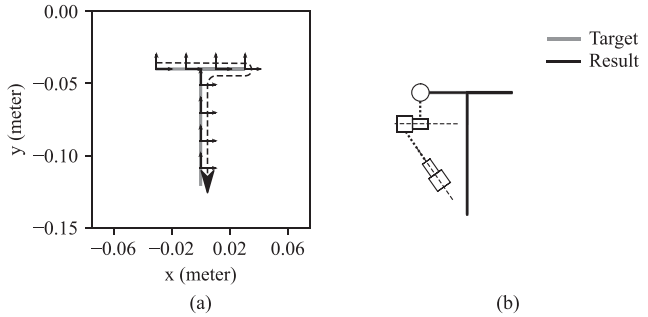

Fig. 7. (a) Target trajectory to design a manipulator to draw the letter "T" on a flat surface without changing the end-effector's orientation. (b) Joint displacements obtained by the optimization of the two joint twists in the initial orientation of the manipulator, target trajectory, and resultant trajectory drawn by the manipulator.

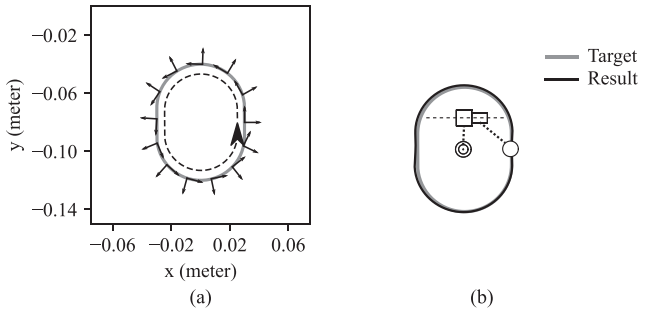

Fig. 8. (a) Target trajectory for designing a manipulator to draw the letter "O" on a flat surface with the rotation of the end-effector. (b) Joint displacements obtained by the optimization of two joint twists in the initial orientation of the manipulator, the target trajectory, and the resultant trajectory drawn by the manipulator.

as in the previous example. Therefore, the objective here is that the displacements of the two joints that realize the approximated trajectory of the end-effectors remain as close as possible to the original trajectory. In this section, we show that the joint displacements for realizing the approximated trajectory can be obtained by using our proposed method.

The obtained twists are as follows:

$$\boldsymbol{\xi}_1 = \begin{bmatrix} 0.0797 \\ -0.0008 \\ 0.0000 \\ 0.0000 \\ 0.0000 \\ -0.9968 \end{bmatrix}, \quad \boldsymbol{\xi}_2 = \begin{bmatrix} 1.0000 \\ 0.0000 \\ 0.0000 \\ 0.0000 \\ 0.0000 \\ 0.0000 \end{bmatrix}. \tag{19}$$

Fig. 7(b) shows the resultant joint displacements. In (19), $\boldsymbol{v}_1^T \boldsymbol{\omega}_1 = 0$ and $\boldsymbol{\omega}_2 = 0$; therefore, they are a revolute joint and a prismatic joint, respectively. Fig. 8(b) also shows the target trajectory and the resultant trajectory generated by the designed manipulator. The drawn path seems close to the target path, but they do not fit perfectly. As a result, the error was relatively larger than that in the previous example; the sum of squared differences was $e = 3.36 \times 10^{-5}$ [J.s].

To show that the obtained joint displacements are better than other displacements, Fig. 9 shows the trajectories obtained by the generalized inverse when we displace the prismatic joint in different angles. Fig. 9(a) and (b) shows the cases in which the prismatic joint is displaced inclining at 90° and 45°, respectively, relative to the optimal displacement around the axis perpendicular to the xy plane. We can see that the error in the path gradually becomes small as the displacement gets closer to the optimal joint displacement. We confirmed that

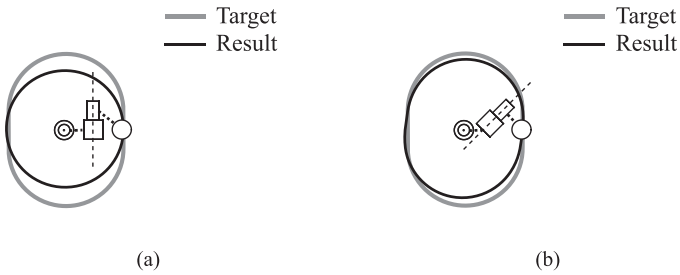

Fig. 9. Trajectories drawn by the manipulator with the prismatic joint inclined with respect to the optimal displacement around the axis perpendicular to the xy plane. (a) 90° inclination. (b) 45° inclination.

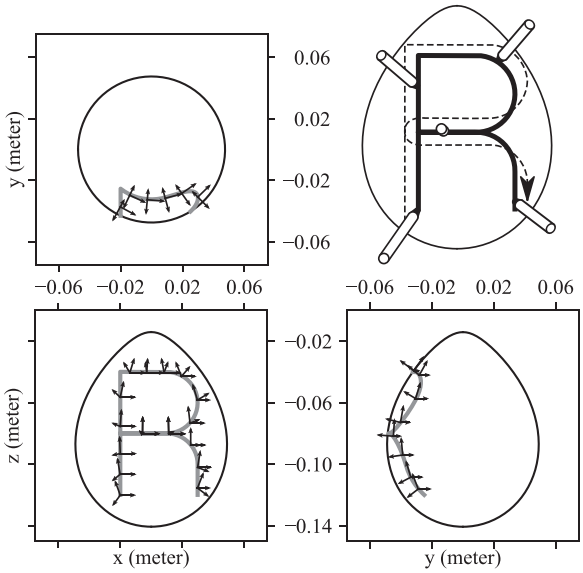

Fig. 10. Letter "R" drawn on an egg shell. The robotic arm to draw the letter "R" on an egg shell was developed by using the proposed methodology for optimizing the design of the joint dispositions.

the error including the orientation also decreases as the displacement gets closer to the optimal value.

C. Drawing "R" on a Plane With Two to Three Joints

The final example is more complex than the previous two examples. Fig. 10 shows the joint displacements designed for drawing the letter "R" on an egg-shaped object by maintaining the end-effector vertical to the surface of the object. We show the optimal joint displacements in the three cases (two screw joints, three screw joints, and an RPR manipulator) in which the joint restrictions were different. The optimal displacement changes according to the restrictions on the types and number of the joints.

1) Optimization Result for Two Joints: First, the twists representing the two joints were optimized. The obtained twists are as follows:

$$\boldsymbol{\xi}_1 = \begin{bmatrix} 0.0000 \\ 0.0004 \\ 0.0008 \\ 0.0012 \\ -0.0004 \\ -1.0000 \end{bmatrix}, \quad \boldsymbol{\xi}_2 = \begin{bmatrix} 0.0453 \\ 0.0763 \\ 0.0190 \\ -0.8556 \\ 0.5097 \\ 0.0025 \end{bmatrix}. \tag{20}$$

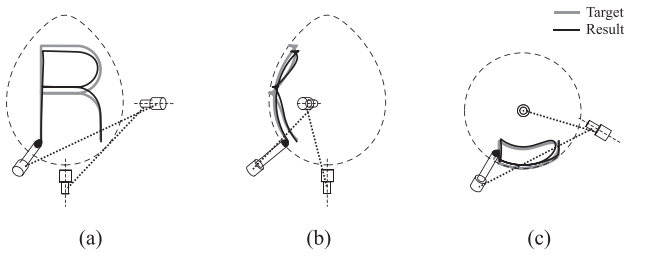

Fig. 11. Joint displacements obtained by the optimization of two joint twists in the initial orientation of the manipulator and resultant trajectories: (a) the front view, (b) the right-side view, and (c) the top view.

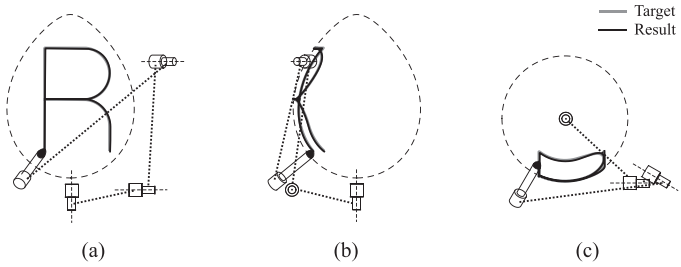

Fig. 12. Joint displacements obtained by the optimization of three joint twists in the initial orientation of the manipulator and the resultant trajectory: (a) the front view, (b) right-side view, and (c) the top view.

The sum of the squared differences between the resultant and the desired trajectories was $e = 3.01 \times 10^{-5}$ [J.s]. Fig. 11 shows the resultant joint displacements in the initial orientation of the manipulator and the trajectory, which was generated when the manipulator followed the desired trajectory based on the generalized inverse represented by (9). It can be seen that the resultant trajectory is close to the desired trajectory, but there are differences between these trajectories.

2) Optimization Result for Three Joints: The twists representing three joints were then optimized, and we obtained the following twists:

$$\boldsymbol{\xi}_1 = \begin{bmatrix} 0.0003 \\ 0.0004 \\ 0.0003 \\ 0.0026 \\ -0.0001 \\ -1.0000 \end{bmatrix}, \boldsymbol{\xi}_2 = \begin{bmatrix} 1.0000 \\ 0.0006 \\ 0.0000 \\ 0.0000 \\ 0.0000 \\ 0.0000 \end{bmatrix}, \boldsymbol{\xi}_3 = \begin{bmatrix} 0.0288 \\ -0.0763 \\ 0.0034 \\ 0.8549 \\ -0.5123 \\ 0.0022 \end{bmatrix}. \tag{21}$$

In this case, the sum of the squared differences was less than that in the previous case by one order of magnitude; it was $e = 3.07 \times 10^{-6}$ [J.s]. The resultant joint displacements in the initial orientation and the resultant trajectory are shown in Fig. 12. The resultant trajectory seems to be very close to the desired trajectory.

We can see that a few components are almost zero. The pitches of the screw motion caused by these twists are $h_1 = -0.00026$, $h_2 = \infty$, and $h_3 = 0.06390$, where h_1, h_2, and h_3 are the pitches of $\boldsymbol{\xi}_1$, $\boldsymbol{\xi}_2$, and $\boldsymbol{\xi}_3$, respectively. This means that the joint represented by $\boldsymbol{\xi}_2$ is a prismatic joint, and the joints represented by $\boldsymbol{\xi}_1$ and $\boldsymbol{\xi}_3$ are close to revolute joints. Therefore, the twists were optimized again under the constraint that $\boldsymbol{\xi}_2$ was a prismatic joint, and that $\boldsymbol{\xi}_1$ and $\boldsymbol{\xi}_3$ were revolute joints, as described in the next section.

3) Constrained Optimization Result for Three Joints: The twists representing the three joints were then optimized under the constraint that the first and the third joints were revolute joints and the second joint was a prismatic joint. Optimization was applied to four parameters in the angular coordinates of a revolute joint and to two parameters in the angular coordinates of a prismatic joint. Therefore, a

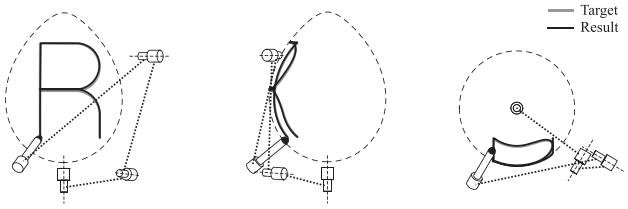

Fig. 13. Joint displacements obtained by the constrained optimization of three joint twists in the initial orientation of the manipulator and the resultant trajectory: (a) the front view, (b) the right-side view, and (c) the top view.

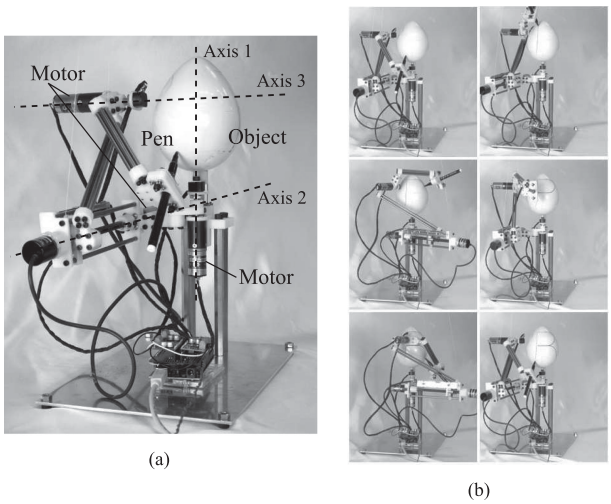

Fig. 14. (a) Manipulator designed based on the optimization result. (b) Sequence of photographs of the manipulator in the process of drawing a letter on an egg-shaped object.

total of 10 parameters represented the twists. The obtained twists were as follows:

$$\boldsymbol{\xi}_1 = \begin{bmatrix} 0.0000 \\ 0.0000 \\ 0.0000 \\ 0.0000 \\ -0.0003 \\ -1.0000 \end{bmatrix}, \boldsymbol{\xi}_2 = \begin{bmatrix} -0.5141 \\ -0.8558 \\ 0.0582 \\ 0.0000 \\ 0.0000 \\ 0.0000 \end{bmatrix}, \boldsymbol{\xi}_3 = \begin{bmatrix} 0.0261 \\ 0.0437 \\ 0.0000 \\ -0.8572 \\ 0.5125 \\ -0.0007 \end{bmatrix}. \quad (22)$$

In this case, the sum of the squared differences was $e = 2.35 \times 10^{-6}$ [J.s]. The error was reduced by constraining the problem because the parameter space in the optimization decreased in size from 15 to 10, and the solver found a better local optimal solution. The resultant joint displacements in the initial orientation and the resultant trajectory are shown in Fig. 13.

D. Implementation

Finally, to validate the optimization result, we developed a manipulator to draw the letter "R" on an egg-shaped object based on the result in Section III-C3. Fig. 14(a) shows the developed manipulator. Axes 1, 2, and 3 in the figure correspond to $\boldsymbol{\xi}_1$, $\boldsymbol{\xi}_2$, and $\boldsymbol{\xi}_3$ in the optimization result, respectively. Axes 1 and 3 are the axes of the revolute joints, and Axis 2 is the axis of a prismatic joint.

As explained in Section II-B, the representation of the joints in the twist does not specify the location of the joint in the specified directions. The end-effector motion caused by the joint is invariant for the shifting in these directions. Specifically, the end-effector motion is invariant for the shifting of a revolute joint along the axial direction and for the shifting of a prismatic joint in any direction. We are not concerned with how those unspecified design parameters, including the link connecting the joints, were determined in this paper. In this example, we designed the manipulator by shifting the joints in their invariant directions and by checking self-collisions and collisions with the environment including the egg-shaped object on the computer-aided design application Autodesk Inventor (Autodesk, Inc., Sausalito, CA, USA).

The revolute joints were driven by dc-geared motors (TG-01H-FU-23-KA, Tsukasa Electric Co., Ltd., Tokyo, Japan), and the prismatic joint was driven by a dc-geared motor (TG-01H-FU-509-KA, Tsukasa Electric Co., Ltd., Tokyo, Japan). The angles of the revolute joints were measured by using potentiometers (CP-2HB, Midori Precisions Co., Ltd., Tokyo, Japan), and the displacements of the prismatic joints were measured by using an optical absolute encoder (E6A2-CW3C, Omron Corporation, Kyoto, Japan). A pen was attached to the endpoint of the manipulator, so that its tip was in contact with the surface of the egg-shaped object fixed to the center of the manipulator. Between the pen and the manipulator, small springs were installed to apply a pressing force on the object.

The manipulator drew the letter by following the discretized trajectory of the position and orientation of the end-effector (defined in the first part of this section). While the manipulator was controlled, the links between the joints were lifted by wires and weights to compensate for the weight of the manipulator and the ease of feedback control. The motors of the manipulator were controlled by using a simple PID controller with feedback from the potentiometers and a rotary encoder.

Fig. 14(b) shows the resultant trajectory of the manipulator when it was controlled to draw the letter "R" on the egg-shaped object. As shown in Fig. 14(b), the manipulator that we developed was able to draw the letter correctly.

IV. Conclusion

In this paper, we proposed an optimization-based methodology to calculate the joint displacement of a serial manipulator to follow an approximate trajectory of an end-effector. Instead of solving various inverse kinematics problems for different points on the trajectory during optimization, we used the generalized differential inverse kinematics to obtain the approximated trajectory realized by joint kinematics. Finally, a few examples of kinematic synthesis by using the proposed method were shown with the design and implementation of the manipulator based on the optimization result.

Unlike other researches on kinematic synthesis, in this paper, we did not specify the points through which the end-effector of a manipulator must pass with a given orientation. In case an appropriate number of such points is given, the kinematic synthesis problem can be solved. However, when only the target trajectory is given, the kinematic synthesis needs to be performed by using some approximation. Therefore, the proposed method is useful in the latter case, especially when the points on the trajectory need to be evaluated equally. Furthermore, the proposed method introduced the differential inverse for the evaluation and realized the optimization of a high-dimensional parameter space at low calculation cost.

This methodology has a few limitations that must be overcome to improve the efficiency and generalize the method. In particular, an efficient and reliable optimization method is required to enhance the proposed methodology. The evaluation function introduced in the method is non-differentiable, and the optimized parameters were represented on a manifold. Therefore, only a very few optimization methods can be applied to our problem, and we used an existing derivative-free optimization methodology without modifying it. If a more appropriate optimization method matching the problem was available, a manipulator design with a smaller error from the target could have been achieved efficiently. The technique of optimization on manifolds has attracted a lot of attention these days [46], [47], although further research needs to be done to apply it to the optimization in this paper. Furthermore, as

we explained in Section II-A, we have only considered the functional aspect of the joints represented as twists, and we have not dealt with their actual positions in the design of a manipulator. The joints' actual positions in the design relate to the size of the links connecting them or the self-collisions of the joints and the links. Ideally, these should also be solved in the optimization process, for example, by transforming the joint twists into the Denavit–Hartenberg parameters with additional parameters and optimizing these parameters considering the constraints on these parameters related to the collisions and the size at the same time. This kind of extension of the proposed method will enhance the versatility of the method. These subjects will be investigated in our future research.

ACKNOWLEDGMENT

The authors would like to thank Enago (www.enago.jp) for the English language review.

REFERENCES

[1] A. Kumar and K. Waldron, "The workspaces of a mechanical manipulator," *J. Mech. Des.*, vol. 103, no. 3, pp. 665–672, 1981.
[2] D. Yang and T. Lee, "On the workspace of mechanical manipulators," *J. Mechanisms, Transm., Autom. Des.*, vol. 105, no. 1, pp. 62–69, 1983.
[3] F. Freudenstein and E. Primrose, "On the analysis and synthesis of the workspace of a three-link, turning-pair connected robot arm," *J. Mechanisms, Transm., Autom. Des.*, vol. 106, no. 3, pp. 365–370, 1984.
[4] Y. Tsai and A. Soni, "The effect of link parameter on the working space of general 3R robot arms," *Mechanism Mach. Theory*, vol. 19, no. 1, pp. 9–16, 1984.
[5] C.-C. D. Lin and F. Freudenstein, "Optimization of the workspace of a three-link turning-pair connected robot arm," *Int. J. Robot. Res.*, vol. 5, no. 2, pp. 104–110, 1986.
[6] R. Vijaykumar, K. Waldron, and M. Tsai, "Geometric optimization of serial chain manipulator structures for working volume and dexterity," *Int. J. Robot. Res.*, vol. 5, no. 2, pp. 91–103, 1986.
[7] B. Paden and S. Sastry, "Optimal kinematic design of 6R manipulators," *Int. J. Robot. Res.*, vol. 7, no. 2, pp. 43–61, 1988.
[8] Y. Tsai and A. Soni, "Workspace synthesis of 3R, 4R, 5R and 6R robots," *Mechanism Mach. Theory*, vol. 20, no. 6, pp. 555–563, 1985.
[9] C. M. Gosselin and M. Guillot, "The synthesis of manipulators with prescribed workspace," *J. Mech. Des.*, vol. 113, no. 4, pp. 451–455, 1991.
[10] J. K. Salisbury and J. J. Craig, "Articulated hands: Force control and kinematic issues," *Int. J. Robot. Res.*, vol. 1, no. 1, pp. 4–17, 1982.
[11] J. Angeles and C. S. López-Cajún, "Kinematic isotropy and the conditioning index of serial robotic manipulators," *Int. J. Robot. Res.*, vol. 11, no. 6, pp. 560–571, 1992.
[12] M. Gonzalez-Palacios, J. Angeles, and F. Ranjbaran, "The kinematic synthesis of serial manipulators with a prescribed Jacobian," in *Proc. IEEE Int. Conf. Robot. Autom.*, 1993, pp. 450–455.
[13] C. A. Klein and B. E. Blaho, "Dexterity measures for the design and control of kinematically redundant manipulators," *Int. J. Robot. Res.*, vol. 6, no. 2, pp. 72–83, 1987.
[14] P. Wenger, "Some guidelines for the kinematic design of new manipulators," *Mechanism Mach. Theory*, vol. 35, no. 3, pp. 437–449, 2000.
[15] T. Yoshikawa, "Manipulability of robotic mechanisms," *Int. J. Robot. Res.*, vol. 4, no. 2, pp. 3–9, 1985.
[16] S. Patel and T. Sobh, "Task based synthesis of serial manipulators," *J. Adv. Res.*, vol. 6, no. 3, pp. 479–492, 2015.
[17] C. J. J. Paredis and P. K. Khosla, "An approach for mapping kinematic task specifications into a manipulator design," in *Proc. 5th Int. Conf. Adv. Robot. 'Robot. Unstructured Environ.*, 1991, pp. 556–561.
[18] C. Valsamos, V. Moulianitis, and N. Aspragathos, "Kinematic synthesis of structures for metamorphic serial manipulators," *J. Mechanisms Robot.*, vol. 6, no. 4, 2014, Art. no. 041005.
[19] S. Manoochehri and A. Seireg, "Computer-aided generation of an optimum machine topology for specified tasks," *J. Comput. Mech. Eng.*, vol. 6, no. 3, pp. 10–24, 1987.
[20] S. Manoochehri and A. A. Seireg, "A computer-based methodology for the form synthesis and optimal design of robot manipulators," *J. Mech. Des.*, vol. 112, pp. 501–508, 1990.
[21] S. Ha, S. Coros, A. Alspach, J. Kim, and K. Yamane, "Computational co-optimization of design parameters and motion trajectories for robotic systems," *Int. J. Robot. Res.*, vol. 37, no. 13-14, pp. 1521–1536, 2018.
[22] C. H. Suh, "On the duality in the existence of RR links for three positions," *J. Eng. Ind.*, vol. 91, no. 1, pp. 129–134, 1969.
[23] J. M. McCarthy, "The synthesis of planar RR and spatial CC chains and the equation of a triangle," *J. Mech. Des.*, vol. 117, no. B, pp. 101–106, 1995.
[24] E. Lee and C. Mavroidis, "Solving the geometric design problem of spatial 3R robot manipulators using polynomial homotopy continuation," *J. Mech. Des.*, vol. 124, no. 4, pp. 652–661, 2002.
[25] E. Lee and C. Mavroidis, "Geometric design of 3R robot manipulators for reaching four end-effector spatial poses," *Int. J. Robot. Res.*, vol. 23, no. 3, pp. 247–254, 2004.
[26] P. Chen and B. Roth, "Design equations for finitely and infinitesimally separated position synthesis of binary links and combined link chains," *J. Eng. Ind.*, vol. 91, no. 1, pp. 209–219, 1968.
[27] H.-J. Su and J. M. McCarthy, "Kinematic synthesis of RPS serial chains," in *Proc. ASME Int. Des. Eng. Tech. Conf. Comput. Inf. Eng. Conf.*, 2003, pp. 1041–1047.
[28] A. Perez-Gracia and J. McCarthy, "Geometric design of RRP, RPR and PRR serial chains," *Mechanism Mach. Theory*, vol. 40, no. 11, pp. 1294–1311, 2005.
[29] E. Singla, S. Tripathi, V. Rakesh, and B. Dasgupta, "Dimensional synthesis of kinematically redundant serial manipulators for cluttered environments," *Robot. Auton. Syst.*, vol. 58, no. 5, pp. 585–595, 2010.
[30] C. W. Wampler, A. P. Morgan, and A. J. Sommese, "Numerical continuation methods for solving polynomial systems arising in kinematics," *J. Mech. Des.*, vol. 112, no. 1, pp. 59–68, 1990.
[31] H.-J. Su, J. M. McCarthy, and L. T. Watson, "Generalized linear product homotopy algorithms and the computation of reachable surfaces," vol. 4, no. 3, pp. 226–234, 2004.
[32] J. M. McCarthy and G. S. Soh, *Geometric Design of Linkages*, vol. 11. New York, NY, USA: Springer-Verlag, 2010.
[33] A. Perez-Gracia and J. M. McCarthy, "Dual quaternion synthesis of constrained robotic systems," *Trans. ASME. J. Mech. Des.*, vol. 126, pp. 425–435, 2004.
[34] A. Perez-Gracia and J. M. McCarthy, "Kinematic synthesis of spatial serial chains using Clifford algebra exponentials," *Proc. Inst. Mech. Eng., Part C, J. Mech. Eng. Sci.*, vol. 220, no. 7, pp. 953–968, 2006.
[35] R. Diankov, "Automated construction of robotic manipulation programs," Ph.D. dissertation, Robot. Inst., Carnegie Mellon Univ., Pittsburgh, PA, USA, 2010.
[36] K. L. Doty, C. Melchiorri, and C. Bonivento, "A theory of generalized inverses applied to robotics," *Int. J. Robot. Res.*, vol. 12, no. 1, pp. 1–19, 1993.
[37] R. M. Murray, Z. Li, and S. S. Sastry, *A Mathematical Introduction to Robotic Manipulation*. Boca Raton, FL, USA: CRC Press, 1994.
[38] O. Khatib, "A unified approach for motion and force control of robot manipulators: The operational space formulation," *IEEE J. Robot. Autom.*, vol. RA-3, no. 1, pp. 43–53, Feb. 1987.
[39] C. A. Klein and C.-H. Huang, "Review of pseudoinverse control for use with kinematically redundant manipulators," *IEEE Trans. Syst., Man Cybern.*, vol. SMC-13, no. 2, pp. 245–250, Mar./Apr. 1983.
[40] L. Saab, O. E. Ramos, F. Keith, N. Mansard, P. Soueres, and J.-Y. Fourquet, "Dynamic whole-body motion generation under rigid contacts and other unilateral constraints," *IEEE Trans. Robot.*, vol. 29, no. 2, pp. 346–362, Apr. 2013.
[41] V. Krovi, G. Ananthasuresh, and V. Kumar, "Kinematic synthesis of spatial RR dyads for path following with applications to coupled serial chain mechanisms," *J. Mech. Des.*, vol. 123, no. 3, pp. 359–366, 2001.
[42] G. Alıcı and B. Shirinzadeh, "Optimum synthesis of planar parallel manipulators based on kinematic isotropy and force balancing," *Robotica*, vol. 22, no. 1, pp. 97–108, 2004.
[43] L. M. Rios and N. V. Sahinidis, "Derivative-free optimization: a review of algorithms and comparison of software implementations," *J. Global Optim.*, vol. 56, no. 3, pp. 1247–1293, 2013.
[44] W. Huyer and A. Neumaier, "Global optimization by multilevel coordinate search," *J. Global Optim.*, vol. 14, no. 4, pp. 331–355, 1999.
[45] A. Neumaier, "MCS: Global optimization by multilevel coordinate search," 2000. [Online]. Available: https://www.mat.univie.ac.at/∼neum/software/mcs/
[46] P.-A. Absil, R. Mahony, and R. Sepulchre, *Optimization Algorithms on Matrix Manifolds*. Princeton, NJ, USA: Princeton Univ. Press, 2009.
[47] S. Traversaro, S. Brossette, A. Escande, and F. Nori, "Identification of fully physical consistent inertial parameters using optimization on manifolds," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2016, pp. 5446–5451.