



Research paper



Forward kinematics of a cable-driven parallel robot with pose estimation error covariance bounds

Samir Patel, Vinh L. Nguyen, Ryan J. Caverly *

Department of Aerospace Engineering and Mechanics, University of Minnesota, 110 Union St., SE, Minneapolis, MN 55455, United States

ARTICLE INFO

Keywords:

Cable-driven parallel robots
Pose estimation
Forward kinematics
Least-squares estimation

ABSTRACT

This paper presents four forward kinematics algorithms for an over-constrained six degree-of-freedom (DOF) cable-driven parallel robot (CDPR) that in addition to computing the end-effector's pose, also provide covariance bounds on the pose estimation error. The first two proposed methods are based on cable-length and cable-length-squared loop-closure equations and the use of unconstrained attitude parameterizations to describe the orientation of the CDPR end-effector. The second pair of methods involve constrained attitude parameterizations and are also based on cable-length and cable-length-squared loop-closure equations. Nonlinear least-squares optimization is used in each of these methods to iteratively compute the forward kinematics solution and determine covariance bounds on the pose estimation error. Attitude identities are used to obtain analytic expressions for the computations whenever possible. The forward kinematics algorithms are validated through Monte-Carlo simulations, where Euler-angle-sequence, quaternion, and DCM parameterizations of the end-effector attitude are implemented and the accuracy of the covariance bounds is demonstrated. It is also shown that the method based on the cable-length-squared loop-closure equations yields improved convergence properties compared to the cable-length loop-closure equations.

1. Introduction

Cable-driven parallel robots (CDPRs) use a set of cables and winches to manipulate a payload or an end-effector. Owing to their large workspace, high-acceleration capabilities, and large payload-to-weight ratio, CDPRs are employed in various applications, such as human body and waist rehabilitation [1], large-scale 3D printing [2], motion simulation [3], and vibration suppression of flexible structures [4]. Knowledge of the end-effector's position and orientation, known as the end-effector's pose, is necessary to alter and control the motion of the end-effector. The primary method to calculate the pose of the end-effector is through forward kinematics, which computes the pose by solving nonlinear geometric loop-closure equations using measurements of the CDPR's cable lengths at a given instance in time. There are various iterative [5–7], interval [8], and hybrid [9] forward kinematics algorithms developed to solve this challenging problem, that in general features multiple solutions. Cable sag [10] and pulley kinematics [11] can be incorporated into these forward kinematic approaches and their calibration, usually with additional computational expense.

Additional sensors can be used to improve the CDPR pose estimate, including an end-effector-mounted inertial measurement unit (IMU) [12–17] or camera [18,19], external vision systems [20–22], cable angle sensors [23,24], and direct cable length measurement sensors [25]. Many of these sensors are most effective when fused with the solution from forward kinematics. Successful implementations of this include Kalman filtering algorithms that fuse forward kinematics with IMU measurements [12–17] and the fusion of cable angle sensors and forward kinematics through the best least unbiased estimator [23,24]. However,

* Corresponding author.

E-mail addresses: samir004@umn.edu (S. Patel), nguy2876@umn.edu (V.L. Nguyen), rcaverly@umn.edu (R.J. Caverly).

sensor fusion algorithms require quantitative knowledge of the accuracy of the sensor measurements being fused together in order to maintain optimality and consistency of the fused pose estimate. In this regard, covariance bounds on the error in the pose estimate obtained through forward kinematics are required when a forward kinematics pose estimate is fused with additional sensor information using Kalman filtering or least-squares estimation [24]. Monte-Carlo simulations or large batches of experiments can be used to quantify this covariance [5], although this is time consuming, especially since the covariance is pose-dependent.

Preliminary work towards an alternative approach was performed in [26], where covariance bounds on the forward kinematics pose estimation error were found through analytic linearizations of two different forms of the loop-closure equations. Although the methods presented in [26] showed promise, they are only derived for use with unconstrained attitude parameterizations (e.g., Euler angle sequence, Rodrigues parameters) and a rigorous investigation into the accuracy of the covariance bounds obtained was not performed. There has been a recent increase in the number of pose estimation and optimization algorithms [27–31] as well as pose controllers [32,33] that make use of constrained attitude parameterizations (e.g., the quaternion, the direction cosine matrix or rotation matrix) to take advantage of the multiplicative nature of attitude errors. In order to be compatible with these approaches, it is important to develop forward kinematics algorithms that can accurately assess the pose estimation error covariance when using constrained attitude parameterizations.

This paper greatly expands upon the work of [26] by deriving forward kinematics algorithms for constrained attitude parameterizations and performing a detailed Monte Carlo analysis to confirm the accuracy of the proposed covariance bound computation with both unconstrained and constrained attitude parameterizations. Constrained attitude parameterizations are included in the proposed forward kinematics algorithms through the use of the identities derived in [34]. These identities allow for an analytic computation of the Jacobian used in the iterative Levenberg–Marquardt algorithm and provide a means to derive approximate covariance bounds in the constrained attitude parameters. To the best of the knowledge of the authors, the novel contributions specific to this paper are summarized as (1) the first CDPR forward kinematics algorithms with constrained attitude parameterizations to involve analytic Jacobian expressions and covariance bounds on the pose estimation error, (2) numerical validation of the accuracy of the pose estimation error covariance bounds obtained with the proposed forward kinematics methods, and (3) a comparison between the use of cable-length-squared loop-closure equations (denoted as Method 1) and cable-length loop-closure equations (denoted as Method 2) when performing CDPR forward kinematics.

The remainder of this paper proceeds as follows. Important notation and a discussion of attitude parameterizations are provided in this section. Section 2 introduces a description of the class of over-constrained CDPRs considered in this work and derives the cable-length and cable-length-squared loop-closure equations. The proposed forward kinematics algorithms with unconstrained and constrained attitude parameterizations are developed in Sections 3 and 4, respectively. Numerical simulation results of the proposed forward kinematic approaches with Euler angle, quaternion, and direction cosine matrix (DCM) attitude representations are presented in Section 5 to analyze the statistical accuracy of the computed covariance bounds and assess the accuracy and computational efficiency of the methods. Concluding remarks are provided in Section 6.

1.1. Notation

Identity and zero matrices are denoted as $\mathbf{1}$ and $\mathbf{0}$, respectively. The reference frame \mathcal{F}_i is defined by a set of three dextral, orthonormal vectors $\underline{i}^1, \underline{i}^2, \underline{i}^3$. The vector \underline{n} resolved in \mathcal{F}_i is written as $\mathbf{n}_i = [n_{i1} \ n_{i2} \ n_{i3}]^\top$. The cross operator is denoted by $(\cdot)^\times : \mathbb{R}^3 \mapsto \mathfrak{so}(3)$ and satisfies the properties $\mathbf{n}^\times \mathbf{m} = -\mathbf{m}^\times \mathbf{n}$ and $\mathbf{z}^{\times^\top} = -\mathbf{z}^\times$, where $\mathbf{n}, \mathbf{m}, \mathbf{z} \in \mathbb{R}^3$ and $\mathfrak{so}(3) = \{\mathbf{S} \in \mathbb{R}^{3 \times 3} \mid \mathbf{S} + \mathbf{S}^\top = \mathbf{0}\}$ [35, pp. 525–526]. The notation $x \sim \mathcal{N}(\bar{x}, \sigma_x^2)$ denotes that x is a continuous random variable that satisfies a normal distribution with mean \bar{x} and variance σ_x^2 .

1.2. Attitude parameterizations

The attitude or orientation of reference frame \mathcal{F}_b relative to reference frame \mathcal{F}_a is described by the direction cosine matrix (DCM) $\mathbf{C}_{ba} \in \mathbb{R}^{3 \times 3}$, where $\mathbf{C}_{ba}^\top \mathbf{C}_{ba} = \mathbf{1}$ and $\det(\mathbf{C}_{ba}) = +1$. The DCM \mathbf{C}_{ba} is related to the rotation matrix \mathbf{R}_{ba} that rotates \mathcal{F}_a to align with \mathcal{F}_b by $\mathbf{C}_{ba} = \mathbf{R}_{ba}^\top$. The DCM can be used to change the reference frame in which a vector is resolved. For example, $\mathbf{r}_b = \mathbf{C}_{ba} \mathbf{r}_a$. The angular velocity of \mathcal{F}_b relative to \mathcal{F}_a and resolved in \mathcal{F}_b is $\boldsymbol{\omega}_b^{ba}$. Additionally, the time derivative of \mathbf{C}_{ba} is described by Poisson's equation, where $\dot{\mathbf{C}}_{ba} = -\boldsymbol{\omega}_b^{ba} \times \mathbf{C}_{ba}$.

Attitude parameterizations, which can be thought of as parameterizations of the DCM, are used in many applications, including CDPR forward kinematics. In this paper, attitude parameterizations are categorized as either unconstrained or constrained. An attitude parameterization is considered to be unconstrained if its parameters are not bound to any constraints. For example, the Euler angle sequence is an unconstrained parameterization, whereas, the quaternion must satisfy a unit-length constraint, and is therefore a constrained parameterization. Consider an unconstrained attitude parameterization, $\mathbf{p}^{ba} \in \mathbb{R}^3$. Examples of unconstrained attitude parameters include Euler angle sequences, the rotation vector, Rodrigues parameters, and modified Rodrigues parameters. An identity for unconstrained parameterizations that will be used when formulating the forward kinematics in Section 3 is given by [34]

$$\frac{\partial}{\partial \mathbf{p}^{ba}} (\mathbf{C}_{ba}^\top \mathbf{u}_b) = -\mathbf{C}_{ba}^\top \mathbf{u}_b^\times \mathbf{S}(\mathbf{p}^{ba}), \quad (1)$$

where $\mathbf{S}(\mathbf{p}^{ba})$ is the kinematic mapping matrix that satisfies $\boldsymbol{\omega}_b^{ba} = \mathbf{S}(\mathbf{p}^{ba}) \dot{\mathbf{p}}^{ba}$ and $\mathbf{u}_b \in \mathbb{R}^3$ is any column matrix. A summary of mapping matrices $\mathbf{S}(\mathbf{p}^{ba})$ for a number of unconstrained attitude parameterizations is found in [26]. Next, consider a constrained

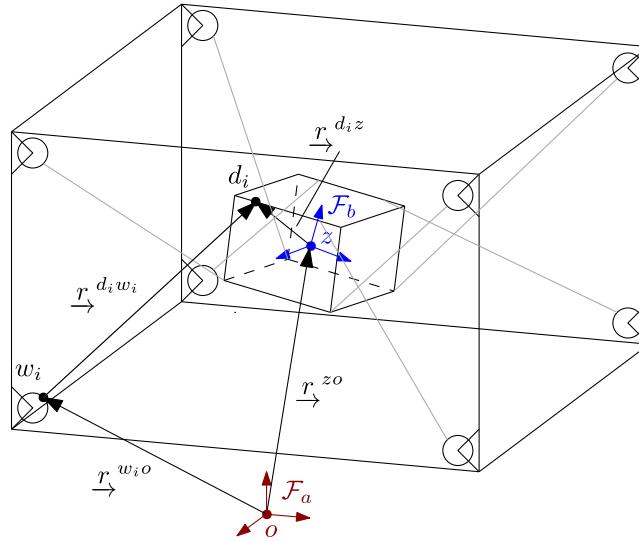


Fig. 1. Schematic of a six-DOF over-constrained CDPR with eight cables.

attitude parameterization, $\mathbf{q}^{ba} \in \mathbb{R}^n$, which must satisfy $\Phi(\mathbf{q}^{ba}) = \mathbf{0}$, where $\Phi(\cdot) : \mathbb{R}^n \rightarrow \mathbb{R}^{n-3}$. Examples of constrained attitude parameterizations include the quaternion (Euler parameters), axis-angle parameters, and the DCM itself. Taking the time derivative of $\Phi(\mathbf{q}^{ba}) = \mathbf{0}$, the attitude parameterization constraint can be expressed at the rate level as $\Xi(\mathbf{q}^{ba})\dot{\mathbf{q}}^{ba} = \mathbf{0}$, where $\Xi(\mathbf{q}^{ba}) = \frac{\partial}{\partial \mathbf{q}^{ba}}(\Phi(\mathbf{q}^{ba}))$. The forward kinematics formulation for constrained attitude parameterizations in Section 4 makes use of the identity [34]

$$\frac{\partial}{\partial \mathbf{q}^{ba}} (\mathbf{C}_{ba}^\top \mathbf{u}_b) \Gamma(\mathbf{q}^{ba}) = -\mathbf{C}_{ba}^\top (\mathbf{q}^{ba}) \mathbf{u}_b^\times, \quad (2)$$

where $\Gamma(\mathbf{q}^{ba})$ satisfies the kinematic relation $\dot{\mathbf{q}}^{ba} = \Gamma(\mathbf{q}^{ba})\omega_b^{ba}$ and $\mathbf{u}_b \in \mathbb{R}^3$ is any column matrix. Substituting $\dot{\mathbf{q}}^{ba} = \Gamma(\mathbf{q}^{ba})\omega_b^{ba}$ into $\Xi(\mathbf{q}^{ba})\dot{\mathbf{q}}^{ba} = \mathbf{0}$ results in $\Xi(\mathbf{q}^{ba})\Gamma(\mathbf{q}^{ba})\omega_b^{ba} = \mathbf{0}$, which must be satisfied for all ω_b^{ba} , leading to the equality $\Xi(\mathbf{q}^{ba})\Gamma(\mathbf{q}^{ba}) = \mathbf{0}$.

2. System description and forward kinematics formulation

This section introduces the CDPR considered in this work, the formulation of the loop-closure equations involved in the proposed forward kinematics algorithms, and a discussion regarding the purpose and scope of the proposed forward kinematics algorithms.

2.1. System description

Consider an over-constrained six-DOF CDPR with $m \geq 7$ cables. Although the forward kinematics formulations in this paper are applicable to any over-constrained CDPR (i.e., CDPRs with $m \geq 7$ cables), a schematic of a six-DOF CDPR with $m = 8$ cables is shown in Fig. 1 for illustrative purposes. An inertial frame F_a and an end-effector-fixed frame F_b are included in Fig. 1, where the inertial frame has an origin point o and the center of mass of the end-effector is at point z . The vector r^{zo} describes the position of z relative to o . Points w_i and d_i denote the winch/anchor and end-effector attachment points of the i th cable, respectively, and $r^{d_i w_i}$ denotes the position of d_i relative to w_i . Additionally, $r^{w_i o}$ defines the position of w_i relative to o , and $r^{d_i z}$ defines the position of d_i relative to z . The CDPR's cables are assumed to experience no sag or elastic deformation, and remain purely in tension. The CDPR need not have a specific geometry (i.e., specific winch/anchor points, end-effector attachment points, and end-effector dimensions), but it is assumed that the position vectors $r^{w_i o}$ and $r^{d_i z}$ are known quantities.

The CDPR's end-effector pose is comprised of the position of the end-effector point z relative to point o , resolved in F_a , as well as the parameterization used to describe the attitude of F_b relative to F_a . For the unconstrained attitude parameterization case, the pose is described by $\rho = [\mathbf{r}_a^{zo\top} \quad \mathbf{p}^{ba\top}]^\top \in \mathbb{R}^6$. Similarly, for the constrained attitude parameterization case, the pose is $\rho = [\mathbf{r}_a^{zo\top} \quad \mathbf{q}^{ba\top}]^\top \in \mathbb{R}^{3+n}$, where n is the dimension of the constrained attitude parameterization $\mathbf{q}^{ba} \in \mathbb{R}^n$.

2.2. Forward kinematics formulation

Forward kinematics involves solving for the pose of the CDPR's end-effector through measurements of the lengths of each cable. The measurement of the i th cable's length is

$$\ell_i = \left\| \mathbf{r}_a^{d_i w_i}(\rho) \right\|_2 + v_{\ell_i}, \quad (3)$$

where $v_{\ell_i} \sim \mathcal{N}(0, \sigma_{\ell_i}^2)$ is zero-mean Gaussian white noise and

$$\mathbf{r}_a^{d_i w_i}(\rho) = \mathbf{r}_a^{zo} + \mathbf{C}_{ba}^T \mathbf{r}_b^{d_i z} - \mathbf{r}_a^{w_i o}$$

is the position of the i th cable's end-effector attachment point relative to its winch or anchor point on the CDPR frame resolved in \mathcal{F}_a , which is derived from the position vector loop-closure relation $\underline{r}^{d_i w_i} = \underline{r}^{zo} + \underline{r}^{d_i z} - \underline{r}^{w_i o}$. The loop-closure equations of all m cables is obtained by concatenating (3) to yield

$$\boldsymbol{\ell} = \mathbf{g}(\rho) + \mathbf{v}_{\ell}, \quad (4)$$

where $\boldsymbol{\ell}^T = [\ell_1 \quad \dots \quad \ell_m]$ contains all of the cable length measurements, $\mathbf{v}_{\ell}^T = [v_1 \quad \dots \quad v_m]$ contains the measurement noise, and the true cable lengths are given by $\mathbf{g}^T(\rho) = [\|\mathbf{r}_a^{d_1 w_1}(\rho)\|_2 \quad \dots \quad \|\mathbf{r}_a^{d_m w_m}(\rho)\|_2]$. The covariance of the measurement noise is defined as $\mathbf{V} = \mathbb{E}[\mathbf{v}_{\ell} \mathbf{v}_{\ell}^T]$.

An alternative description of the loop-closure equations is found by squaring (3) and rearranging the result to obtain

$$\begin{aligned} f_i(\rho, v_{\ell_i}) &\stackrel{\Delta}{=} (\|\mathbf{r}_a^{d_i w_i}(\rho)\|_2 + v_{\ell_i})^2 - \ell_i^2 \\ &= \mathbf{r}_a^{d_i w_i^T}(\rho) \mathbf{r}_a^{d_i w_i}(\rho) + 2 \|\mathbf{r}_a^{d_i w_i}(\rho)\|_2 v_{\ell_i} + v_{\ell_i}^2 - \ell_i^2. \end{aligned} \quad (5)$$

Concatenating (5) for all m cables results in

$$\mathbf{f}(\rho, \mathbf{v}_{\ell}) = \mathbf{0}, \quad (6)$$

where $\mathbf{f}^T(\rho, \mathbf{v}_{\ell}) = [f_1(\rho, v_{\ell_1}) \quad \dots \quad f_m(\rho, v_{\ell_m})]$.

For the remainder of the paper, (4) is referred to as the cable-length loop-closure equations and (6) is referred to as the cable-length-squared loop-closure equations.

2.3. Discussion

The forward kinematics formulation outlined in Section 2.2 is concerned with making use of cable length measurements and loop-closure equations (either cable-length or cable-length-squared loop-closure equations) to estimate the pose of the CDPR's end-effector. In the absence of measurement noise, there can be a number of solutions to a CDPR's forward kinematics. Typically, there are no exact forward kinematics solutions in the presence of measurement noise, but there may be a multiple candidate solutions that are equally consistent with the loop-closure equations (i.e., multiple CDPR end-effector poses satisfy (4) and/or (6) within similar numerical tolerances). The goal of the forward kinematics algorithms proposed in this paper is to determine a single likely solution in a computationally-efficient manner, and is not concerned with determining all possible solutions. Although there is great merit in determining all possible forward kinematics solutions (e.g., the interval analysis method in [36]), the authors are motivated more by the need to determine a single CDPR end-effector pose for use inside a real-time sensor fusion framework or feedback controller. With this in mind, the forward kinematics algorithms considered in this work are considered local approaches and are designed to determine a solution within some neighborhood of an initial pose estimate.

The proposed forward kinematics algorithms are intended to be applicable to a CDPR regardless of its end-effector motion or acceleration profile, which limits the use of additional measurements or model information. For example, cable tensions and geometrico-static equations could be used in addition to cable length measurements within a forward kinematics algorithm in the case of a static or quasi-static CDPR end-effector [37]. Such approaches can be very useful in situations where the CDPR end-effector does not experience high accelerations, but the geometrico-static equations do not hold when the end-effector undergoes larger accelerations. The CDPR's dynamic equations of motion can be used in applications with larger accelerations, but this is more complicated, as it requires measurements or estimates of the end-effector's velocity and/or acceleration. Dynamics-based CDPR pose estimation is amenable to Kalman filtering approaches, which have shown promise in a number of studies [12–16], but this is far-removed from what is traditionally described as CDPR forward kinematics (i.e., the computation of an end-effector pose given cable length measurements). To this end, the proposed forward kinematics algorithms aim to be generally applicable to CDPRs regardless of their acceleration, with real-time capability, while only making use of cable length measurements. A discussion on possible extensions of these approaches to incorporate additional measurements and/or different models is found in Section 3.3.

3. Forward kinematics with unconstrained attitude parameterizations

This section presents two forward kinematics methods based on the use of an unconstrained attitude parameterization to describe the pose of the CDPR's end-effector.

3.1. Method 1: Cable-length-squared loop-closure equations

Linearizing the cable-length-squared loop-closure equation in (6) about the pose $\bar{\rho}$ and zero mean noise $\mathbb{E}[\mathbf{v}_\ell] = \mathbf{0}$ yields

$$\mathbf{0} = \mathbf{f}(\rho, \mathbf{v}_\ell) \approx \mathbf{f}(\bar{\rho}, \mathbf{0}) + \mathbf{J}_\rho(\bar{\rho})\delta\rho + \mathbf{J}_v(\bar{\rho})\mathbf{v}_\ell, \quad (7)$$

where $\delta\rho = \rho - \bar{\rho}$, $\mathbf{J}_\rho(\rho) = \frac{\partial \mathbf{f}(\rho, \mathbf{0})}{\partial \rho}$, $\mathbf{J}_v(\rho) = \frac{\partial \mathbf{f}}{\partial \mathbf{v}_\ell}$, and

$$\mathbf{f}(\bar{\rho}, \mathbf{0}) = \mathbb{E}[\mathbf{f}(\bar{\rho}, \mathbf{v}_\ell)] = \begin{bmatrix} \left\| \mathbf{r}_a^{d_1 w_1}(\bar{\rho}) \right\|_2^2 + \sigma_{\ell_1}^2 - \ell_1^2 \\ \vdots \\ \left\| \mathbf{r}_a^{d_m w_m}(\bar{\rho}) \right\|_2^2 + \sigma_{\ell_m}^2 - \ell_m^2 \end{bmatrix}. \quad (8)$$

Note that expression for $\mathbf{f}(\bar{\rho}, \mathbf{0})$ in (8) makes use of the fact that for the zero-mean Gaussian white noise $v_{\ell_i} \sim \mathcal{N}(0, \sigma_{\ell_i}^2)$, implying that $\mathbb{E}[v_{\ell_i}^2] = \sigma_{\ell_i}^2$ [38, p. 20], which was unaccounted for in the preliminary work of [26]. The Jacobian $\mathbf{J}_v(\rho)$ is expressed as $\mathbf{J}_v(\rho) = \text{diag}\{2\left\| \mathbf{r}_a^{d_1 w_1}(\rho) \right\|_2^2, \dots, 2\left\| \mathbf{r}_a^{d_m w_m}(\rho) \right\|_2^2\}$, while the Jacobian $\mathbf{J}_\rho(\rho)$ is solved for as

$$\mathbf{J}_\rho(\rho) = \begin{bmatrix} 2\mathbf{r}_a^{d_1 w_1^\top}(\rho) \begin{bmatrix} \mathbf{1} & \frac{\partial}{\partial \mathbf{p}^{ba}} (\mathbf{C}_{ba}^\top(\mathbf{p}^{ba}) \mathbf{r}_b^{d_1 z}) \end{bmatrix} \\ \vdots \\ 2\mathbf{r}_a^{d_m w_m^\top}(\rho) \begin{bmatrix} \mathbf{1} & \frac{\partial}{\partial \mathbf{p}^{ba}} (\mathbf{C}_{ba}^\top(\mathbf{p}^{ba}) \mathbf{r}_b^{d_m z}) \end{bmatrix} \end{bmatrix}. \quad (9)$$

The identity in (1) is used to simplify (9) to the equivalent analytic expression

$$\mathbf{J}_\rho(\rho) = \begin{bmatrix} 2\mathbf{r}_a^{d_1 w_1^\top}(\rho) \begin{bmatrix} \mathbf{1} & -\mathbf{C}_{ba}^\top(\mathbf{p}^{ba}) \mathbf{r}_b^{d_1 z^\times} \mathbf{S}(\mathbf{p}^{ba}) \end{bmatrix} \\ \vdots \\ 2\mathbf{r}_a^{d_m w_m^\top}(\rho) \begin{bmatrix} \mathbf{1} & -\mathbf{C}_{ba}^\top(\mathbf{p}^{ba}) \mathbf{r}_b^{d_m z^\times} \mathbf{S}(\mathbf{p}^{ba}) \end{bmatrix} \end{bmatrix}.$$

The forward kinematics problem is formulated as a nonlinear least-squares optimization problem. Intuitively, the goal is to find the pose $\rho \in \mathbb{R}^6$ that is most compatible with the nonlinear loop-closure equation in (6). Due to the presence of noise in the cable length measurements, it is known that an exact solution to (6) will not be found and instead a least-squares solution is desired. Mathematically, this optimization problem is described by

$$\min_{\rho \in \mathbb{R}^6} \mathbf{f}^\top(\rho, \mathbf{v}_\ell) \mathbf{W}^{-1}(\rho) \mathbf{f}(\rho, \mathbf{v}_\ell), \quad (10)$$

where the weighting matrix $\mathbf{W}(\rho)$ is defined as

$$\mathbf{W}(\rho) = \text{diag}\{4\sigma_{\ell_1}^2 \left\| \mathbf{r}_a^{d_1 w_1}(\rho) \right\|_2^2, \dots, 4\sigma_{\ell_m}^2 \left\| \mathbf{r}_a^{d_m w_m}(\rho) \right\|_2^2\}.$$

The matrix $\mathbf{W}(\rho)$ captures the linear covariance in the individual cable-length-squared measurement channels due to the presence of \mathbf{v}_ℓ in the cable length measurements. Specifically, $\mathbf{W}(\rho)$ is derived as

$$\begin{aligned} \mathbf{W}(\rho) &= \mathbb{E}[(\mathbf{J}_v(\rho)\mathbf{v}_\ell)(\mathbf{J}_v(\rho)\mathbf{v}_\ell)^\top] \\ &= \mathbf{J}_v(\rho) \mathbb{E}[\mathbf{v}_\ell \mathbf{v}_\ell^\top] \mathbf{J}_v^\top(\rho) \\ &= \mathbf{J}_v(\rho) \mathbf{V} \mathbf{J}_v^\top(\rho) \\ &= \text{diag}\{4\sigma_{\ell_1}^2 \left\| \mathbf{r}_a^{d_1 w_1}(\rho) \right\|_2^2, \dots, 4\sigma_{\ell_m}^2 \left\| \mathbf{r}_a^{d_m w_m}(\rho) \right\|_2^2\}. \end{aligned}$$

Note that the linearization of $\mathbf{f}(\rho, \mathbf{v}_\ell)$ in (7) is used when computing this weighting, as this covariance is a linear property and requires a linearization. The inclusion of the weighting $\mathbf{W}^{-1}(\rho)$ in the objective function is an important part of the optimization problem, as it emphasizes the importance of minimizing errors in the channels of $\mathbf{f}(\rho, \mathbf{v}_\ell)$ that are less affected by measurement noise. For example, if the first diagonal entry of $\mathbf{W}(\rho)$ is ten times smaller than its second diagonal entry, then the objective function in (10) indicates that it is ten times more important to minimize the square of the error in the first entry of $\mathbf{f}(\rho, \mathbf{v}_\ell)$ compared to its second entry. The solution to this optimization problem with the weighting $\mathbf{W}^{-1}(\rho)$ also has the benefit of being the minimum variance solution [39, Section 2.1].

In practice, it is reasonable to assume that all winch encoders have the same noise properties and thus, $\sigma_{\ell_i} = \sigma_\ell$, $i = 1, \dots, m$, which leads to the simplifications $\mathbf{W}(\rho) = 4\sigma_\ell^2 \text{diag}\{\left\| \mathbf{r}_a^{d_1 w_1}(\rho) \right\|_2^2, \dots, \left\| \mathbf{r}_a^{d_m w_m}(\rho) \right\|_2^2\}$ and $\mathbf{V} = \sigma_\ell^2 \mathbf{1}$. To maintain the generality of the proposed algorithms to varying noise properties, this simplification is not assumed in the derivation of the proposed forward kinematics methods.

The Levenberg–Marquardt algorithm [39–41] is used to solve the optimization problem in (10) by iteratively solving for perturbations about $\bar{\rho}$ and making use of the linearization in (7) to obtain

$$\min_{\delta\rho \in \mathbb{R}^6} (\mathbf{f}(\bar{\rho}, \mathbf{0}) + \mathbf{J}_\rho(\bar{\rho})\delta\rho)^\top \mathbf{W}^{-1}(\bar{\rho}) (\mathbf{f}(\bar{\rho}, \mathbf{0}) + \mathbf{J}_\rho(\bar{\rho})\delta\rho). \quad (11)$$

Note that the term $\mathbf{J}_v(\bar{\rho})\mathbf{v}_\ell$ in the linearization of (7) is not present in (11), as \mathbf{v}_ℓ is unmeasurable noise that is zero mean, and thus, can be assumed to not affect the objective function in (11) when averaged across multiple samples. This is a standard practice when formulating nonlinear least-squares optimization problems [39, pp. 24–28]. The Levenberg–Marquardt algorithm involves selecting an initial pose estimate ρ_0 and solving (11) at the j th iteration to generate the update law

$$\delta\rho = -(\mathbf{J}_\rho^\top(\rho_j)\mathbf{W}^{-1}(\rho_j)\mathbf{J}_\rho(\rho_j) + \eta\mathbf{1})^{-1}\mathbf{J}_\rho^\top(\rho_j)\mathbf{W}^{-1}(\rho_j)\mathbf{f}(\rho_j, \mathbf{0}),$$

where $\rho_{j+1} = \rho_j + \delta\rho$ and the parameter $\eta > 0$ provides numerical damping to the iterative calculations. The damping parameter can be chosen as a constant or be normalized to the values of $\mathbf{J}_\rho^\top(\rho_j)\mathbf{W}^{-1}(\rho_j)\mathbf{J}_\rho(\rho_j)$ at each iteration [41]. The iteration is terminated when a specified stopping criterion is satisfied, such as achieving convergence in terms of $\|\delta\rho\|_2 < s_1$ or when a maximum number of iterations is reached, where $s_1 > 0$ is a numerical tolerance.

Through the use of least-squares optimization theory, the covariance of the pose estimation error at the j th iteration is computed as [39, p. 84]

$$\mathbf{P}_j = \mathbb{E}[(\rho_{\text{true}} - \rho_j)(\rho_{\text{true}} - \rho_j)^\top] = (\mathbf{J}_\rho^\top(\rho_j)\mathbf{W}^{-1}(\rho_j)\mathbf{J}_\rho(\rho_j))^{-1},$$

where $\rho_{\text{true}} \in \mathbb{R}^6$ is the true end-effector pose.

3.2. Method 2: Cable-length loop-closure equations

Linearizing the cable-length loop-closure equations in (4) about $\bar{\rho}$ yields the expression

$$\boldsymbol{\ell} \approx \mathbf{g}(\bar{\rho}) + \mathbf{H}(\bar{\rho})\delta\rho + \mathbf{v}_\ell, \quad (12)$$

where $\mathbf{H}(\rho) = \frac{\partial \mathbf{g}}{\partial \rho}$ is computed as

$$\mathbf{H}(\rho) = \begin{bmatrix} \frac{\mathbf{r}_a^{d_1 w_1}(\rho)}{\|\mathbf{r}_a^{d_1 w_1}(\rho)\|_2} & \begin{bmatrix} \mathbf{1} & \frac{\partial}{\partial \mathbf{p}^{ba}}(\mathbf{C}_{ba}^\top(\mathbf{p}^{ba})\mathbf{r}_b^{d_1 z}) \end{bmatrix} \\ \vdots & \\ \frac{\mathbf{r}_a^{d_m w_m}(\rho)}{\|\mathbf{r}_a^{d_m w_m}(\rho)\|_2} & \begin{bmatrix} \mathbf{1} & \frac{\partial}{\partial \mathbf{p}^{ba}}(\mathbf{C}_{ba}^\top(\mathbf{p}^{ba})\mathbf{r}_b^{d_m z}) \end{bmatrix} \end{bmatrix}. \quad (13)$$

In order to obtain an analytic form of the partial derivative in (13), the identity in (1) is implemented, yielding

$$\mathbf{H}(\rho) = \begin{bmatrix} \frac{\mathbf{r}_a^{d_1 w_1}(\rho)}{\|\mathbf{r}_a^{d_1 w_1}(\rho)\|_2} & \begin{bmatrix} \mathbf{1} & -\mathbf{C}_{ba}^\top(\mathbf{p}^{ba})\mathbf{r}_b^{d_1 z^\times} \mathbf{S}(\mathbf{p}^{ba}) \end{bmatrix} \\ \vdots & \\ \frac{\mathbf{r}_a^{d_m w_m}(\rho)}{\|\mathbf{r}_a^{d_m w_m}(\rho)\|_2} & \begin{bmatrix} \mathbf{1} & -\mathbf{C}_{ba}^\top(\mathbf{p}^{ba})\mathbf{r}_b^{d_m z^\times} \mathbf{S}(\mathbf{p}^{ba}) \end{bmatrix} \end{bmatrix}.$$

It is interesting to note that taking the time derivative of (12) and assuming that the measurement noise is negligible results in $\dot{\boldsymbol{\ell}} \approx \mathbf{H}(\bar{\rho})\delta\rho$. This demonstrates that $\mathbf{H}(\bar{\rho})$ is the Jacobian mapping from end-effector pose rates to cable length rates evaluated at $\bar{\rho}$. Furthermore, $\mathbf{H}^\top(\bar{\rho})$ is the Jacobian that maps cable forces to an end-effector wrench [42, pp. 122–124]. A potential benefit of Method 2 is the fact that this Jacobian may already be computed within the CDPR’s software for other purposes (e.g., force distribution).

The forward kinematics problem is formulated as the nonlinear least-squares optimization problem

$$\min_{\rho \in \mathbb{R}^6} (\boldsymbol{\ell} - \mathbf{g}(\rho))^\top \mathbf{V}^{-1} (\boldsymbol{\ell} - \mathbf{g}(\rho)), \quad (14)$$

where $\mathbf{V} = \mathbb{E}[\mathbf{v}_\ell \mathbf{v}_\ell^\top] = \text{diag}(\sigma_{\ell_1}^2, \dots, \sigma_{\ell_m}^2)$ is again the cable length noise covariance. As in Method 1, the goal of this optimization problem is to find the pose $\rho \in \mathbb{R}^6$ that is most compatible with the nonlinear loop-closure equation in (4) in a least-squares sense. The inclusion of the weight \mathbf{V}^{-1} in the objective function is motivated using the same reasoning discussed in Method 1 in order to obtain the minimum variance pose estimate. The noise term \mathbf{v}_ℓ is ignored in the objective function due to the fact that it is zero-mean unmeasurable noise.

The Levenberg–Marquardt algorithm is used to solve the optimization problem in (14) by iteratively solving for perturbations about $\bar{\rho}$ and making use of the linearization in (12) to obtain

$$\min_{\delta\rho \in \mathbb{R}^6} (\boldsymbol{\ell} - \mathbf{g}(\bar{\rho}) + \mathbf{H}(\bar{\rho})\delta\rho)^\top \mathbf{V}^{-1} (\boldsymbol{\ell} - \mathbf{g}(\bar{\rho}) + \mathbf{H}(\bar{\rho})\delta\rho). \quad (15)$$

The Levenberg–Marquardt algorithm starts with the initial pose estimate ρ_0 and proceeds at the j th iteration with the solution to (15), given by

$$\delta\rho = (\mathbf{H}^\top(\rho_j)\mathbf{V}^{-1}\mathbf{H}(\rho_j) + \eta\mathbf{1})^{-1}\mathbf{H}^\top(\rho_j)\mathbf{V}^{-1}(\boldsymbol{\ell} - \mathbf{g}(\rho_j)),$$

where $\rho_{j+1} = \rho_j + \delta\rho$ and $\eta > 0$ is a numerical damping parameter. Similar to Method 1, various stopping criteria can be employed to determine when to terminate the iteration, such as a tolerance on the norm of $\delta\rho$ or a maximum number of iterations. Following

the same least-squares optimization theory used in Method 1, the covariance of the pose estimation error at the j th iteration is computed as [39, p. 84]

$$\mathbf{P}_j = \mathbb{E}[(\rho_{\text{true}} - \rho_j)(\rho_{\text{true}} - \rho_j)^T] = (\mathbf{H}^T(\rho_j)\mathbf{V}^{-1}\mathbf{H}(\rho_j))^{-1}.$$

Algorithm 1 Forward Kinematics Method 1 & Method 2 with Unconstrained Attitude Parameterizations

```

1: Measure  $\ell$ 
2: Set  $\rho_0$ ,  $j = 0$ 
3: while Stopping criteria not met do
4:   if Method 1 then
5:     Calculate  $\mathbf{f} = \mathbf{f}(\rho_j, \mathbf{0})$ ,  $\mathbf{J}_\rho = \mathbf{J}_\rho(\rho_j)$ ,  $\mathbf{W} = \mathbf{W}(\rho_j)$ 
6:      $\rho_{j+1} = \rho_j - (\mathbf{J}_\rho^T \mathbf{W}^{-1} \mathbf{J}_\rho + \eta \mathbf{I})^{-1} \mathbf{J}_\rho^T \mathbf{W}^{-1} \mathbf{f}$ 
7:   else if Method 2 then
8:     Calculate  $\mathbf{g} = \mathbf{g}(\rho_j)$ ,  $\mathbf{H} = \mathbf{H}(\rho_j)$ 
9:      $\rho_{j+1} = \rho_j + (\mathbf{H}^T \mathbf{V}^{-1} \mathbf{H} + \eta \mathbf{I})^{-1} \mathbf{H}^T \mathbf{V}^{-1} (\ell - \mathbf{g})$ 
10:  end if
11:  Set  $j = j + 1$ 
12: end while
13: if Method 1 then
14:   Calculate  $\mathbf{P}_j = (\mathbf{J}_\rho^T \mathbf{W}^{-1} \mathbf{J}_\rho)^{-1}$ 
15: else if Method 2 then
16:   Calculate  $\mathbf{P}_j = (\mathbf{H}^T \mathbf{V}^{-1} \mathbf{H})^{-1}$ 
17: end if
18: Return  $\rho_j$  and  $\mathbf{P}_j$ 
  
```

3.3. Discussion

A summary of the proposed forward kinematics methods for unconstrained attitude parameterizations is given in Algorithm 1. The key difference between Method 1 and Method 2 lies in the loop-closure equations. The cable length measurement noise appears linearly in the cable-length loop-closure equations of Method 2, whereas the measurement noise term appears nonlinearly in the cable-length-squared loop-closure equations of Method 1. This necessitates an additional linearization and complicates the weighting term used to formulate the nonlinear least-squares optimization problem and the covariance of the pose estimation error.

The output of both Method 1 and Method 2 is a pose estimate ρ_j and a pose estimation error covariance matrix \mathbf{P}_j . It is worth reiterating that although there are other forward kinematics algorithms in the literature that can be used to obtain a pose estimate, the authors are not aware of any other methods in the literature that also compute the covariance of the pose estimation error, except for the preliminary work of [26]. The benefits of knowing the covariance of the pose estimation error in addition to the pose estimate are twofold: the covariance provides additional insight into how much the pose estimate can be trusted (e.g., it provides σ bounds on the pose estimate) and it allows for the pose estimate to be optimally fused with other sensor information within a Bayesian framework (e.g., forward kinematics can be incorporated within a Kalman filter).

Due to the possibility of multiple solutions to the CDPR forward kinematics problem, there is no guarantee that the proposed methods converge to the actual end-effector pose. In practice, this is mitigated by using the pose estimate from the previous time step to warm start the Levenberg–Marquardt iteration at the next time step and using a small enough time step. Other warm start strategies, such as the one in [5], can also be implemented to assist with the iterative nature of the solver.

The proposed forward kinematics algorithms are constructed based on loop-closure equations in (4) and (6) that assume the CDPR's cables are inelastic and do not sag. In practice, these effects may be non-negligible and greatly affect the accuracy of the computed pose estimate and pose estimation error covariance. Although this is a limitation of the proposed forward kinematics algorithms in their presented form, in principle, they could be applied to more realistic and sophisticated loop-closure or geometrico-static equations. In particular, Method 1 could be applied to any CDPR loop-closure equations of the form $\mathbf{f}(\rho, \mathbf{v}) = \mathbf{0}$, provided the Jacobians $\mathbf{J}_\rho(\rho) = \frac{\partial \mathbf{f}(\rho, \mathbf{0})}{\partial \rho}$ and $\mathbf{J}_v(\rho) = \frac{\partial \mathbf{f}}{\partial \mathbf{v}}$ can be computed. For example, the catenary-pulley equations in [43] could be used with Method 1 if the measurement noise on the unstretched cable length and cable force is known. Extensions to higher-fidelity models such as the catenary-pulley model in [43] or the geometrico-static model in [37] will be investigated as future follow on work to the foundational theory provided in this paper.

4. Forward kinematics with constrained attitude parameterizations

This section introduces two forward kinematics computations for use with constrained parameterizations of the end-effector attitude.

4.1. Method 1: Cable-length-squared loop-closure equations

In a similar manner to Method 1 for unconstrained attitude parameterizations in Section 3.1, a solution to the nonlinear cable-length-squared loop-closure equations in (6) is desired. Solving the loop-closure equations is more complicated in this case due to the attitude parameterization constraint, which requires that the solution satisfies the constraint $\Phi(\mathbf{q}^{ba}) = \mathbf{0}$.

The nonlinearity of the loop-closure equations is accounted for by linearizing (6) to obtain (7), whose terms are defined the same as in Section 3, except the perturbation in pose is $\delta\rho^T = [\delta\mathbf{r}_a^{zo^T} \quad \delta\mathbf{q}^{ba^T}]$ and the constrained attitude parameterization \mathbf{q}^{ba} replaces the unconstrained attitude parameterization \mathbf{p}^{ba} in the expression for $\mathbf{J}_\rho(\rho)$ in (9). Note that the identity in (1) only applies to unconstrained attitude parameterizations and cannot be used to simplify the expression in (9). The attitude constraint equation $\Phi(\mathbf{q}^{ba}) = \mathbf{0}$ is also linearized to obtain

$$\mathbf{0} = \Phi(\mathbf{q}^{ba}) \approx \Phi(\bar{\mathbf{q}}^{ba}) + \Xi(\bar{\mathbf{q}}^{ba})\delta\mathbf{q}^{ba}, \quad (16)$$

where $\Xi(\bar{\mathbf{q}}^{ba}) = \frac{\partial}{\partial\mathbf{q}^{ba}}(\Phi(\mathbf{q}^{ba}))$. Assuming that $\Phi(\mathbf{q}^{ba}) = \mathbf{0}$ is linearized about an attitude parameterization value $\bar{\mathbf{q}}^{ba}$ that satisfies the attitude parameterization constraint $\Phi(\bar{\mathbf{q}}^{ba}) = \mathbf{0}$, (16) results in $\Xi(\bar{\mathbf{q}}^{ba})\delta\mathbf{q}^{ba} \approx \mathbf{0}$. This can be rewritten as

$$\mathbf{0} \approx \hat{\Xi}(\bar{\mathbf{q}}^{ba})\delta\rho, \quad (17)$$

where $\hat{\Xi}(\bar{\mathbf{q}}^{ba}) = [\mathbf{0} \quad \Xi(\bar{\mathbf{q}}^{ba})]$. Combining (7) and (17) yields

$$\begin{bmatrix} -\mathbf{f}(\bar{\rho}, \mathbf{0}) - \mathbf{J}_v(\bar{\rho})\mathbf{v}_\ell \\ \mathbf{0} \end{bmatrix} \approx \begin{bmatrix} \mathbf{J}_\rho(\bar{\rho}) \\ \hat{\Xi}(\bar{\mathbf{q}}^{ba}) \end{bmatrix} \delta\rho, \quad (18)$$

which are the set of linearized equations that approximate the loop-closure equations and the attitude parameterization constraint. It is possible to directly make use of (18) to solve for $\delta\rho$ given an initial choice or guess of $\bar{\rho}$, however, this is not ideal, as the expression for \mathbf{J}_ρ contains partial derivatives that must be computed and only a linearization of the attitude constraint is included, which means the resulting attitude estimate $\mathbf{q}^{ba} = \bar{\mathbf{q}}^{ba} + \delta\mathbf{q}^{ba}$ may not satisfy $\Phi(\mathbf{q}^{ba}) = \mathbf{0}$. To remedy these issues, the identity $\delta\mathbf{q}^{ba} \approx \Gamma(\bar{\mathbf{q}}^{ba})\delta\psi$ is introduced, where $\delta\psi \in \mathbb{R}^3$ is the differential rotation vector and it is assumed that $\delta\mathbf{q}^{ba}$ is a small perturbation from $\bar{\mathbf{q}}^{ba}$ [34]. As described in Section 1.2, $\Gamma(\mathbf{q}^{ba})$ satisfies the kinematic relation $\dot{\mathbf{q}}^{ba} = \Gamma(\mathbf{q}^{ba})\omega_b^{ba}$ and lies in the nullspace of the constraint matrix $\Xi(\mathbf{q}^{ba})$, resulting in $\Xi(\mathbf{q}^{ba})\Gamma(\mathbf{q}^{ba}) = \mathbf{0}$. The perturbation $\delta\rho$ is rewritten as

$$\delta\rho \approx \hat{\Gamma}(\bar{\mathbf{q}}^{ba})\delta\hat{\rho}, \quad (19)$$

where $\hat{\Gamma}(\bar{\mathbf{q}}^{ba}) = \text{diag}\{\mathbf{1}, \Gamma(\bar{\mathbf{q}}^{ba})\}$ and $\delta\rho^T = [\delta\mathbf{r}_a^{zo^T} \quad \delta\psi^T]$. Substituting (19) into (18) yields

$$\begin{bmatrix} -\mathbf{f}(\bar{\rho}, \mathbf{0}) - \mathbf{J}_v(\bar{\rho})\mathbf{v}_\ell \\ \mathbf{0} \end{bmatrix} \approx \begin{bmatrix} \mathbf{J}_\rho(\bar{\rho})\hat{\Gamma}(\bar{\mathbf{q}}^{ba}) \\ \hat{\Xi}(\bar{\mathbf{q}}^{ba})\hat{\Gamma}(\bar{\mathbf{q}}^{ba}) \end{bmatrix} \delta\hat{\rho}, \quad (20)$$

which simplifies to

$$-\mathbf{f}(\bar{\rho}, \mathbf{0}) - \mathbf{J}_v(\bar{\rho})\mathbf{v}_\ell \approx \hat{\mathbf{J}}_\rho(\bar{\rho})\delta\hat{\rho}, \quad (21)$$

where $\hat{\Xi}(\bar{\mathbf{q}}^{ba})\hat{\Gamma}(\bar{\mathbf{q}}^{ba}) = [\mathbf{0} \quad \Xi(\bar{\mathbf{q}}^{ba})\Gamma(\bar{\mathbf{q}}^{ba})] = \mathbf{0}$ and $\hat{\mathbf{J}}_\rho(\rho) = \mathbf{J}_\rho(\rho)\hat{\Gamma}(\bar{\mathbf{q}}^{ba})$. The constrained attitude parameterization identity in (2) is applied to obtain the analytic expression

$$\hat{\mathbf{J}}_\rho(\rho, \mathbf{0}) = \begin{bmatrix} 2\mathbf{r}_a^{d_1 u_1^T}(\rho) \begin{bmatrix} \mathbf{1} & -\mathbf{C}_{ba}^\top(\mathbf{q}^{ba})\mathbf{r}_b^{d_1 z^X} \end{bmatrix} \\ \vdots \\ 2\mathbf{r}_a^{d_m u_m^T}(\rho) \begin{bmatrix} \mathbf{1} & -\mathbf{C}_{ba}^\top(\mathbf{q}^{ba})\mathbf{r}_b^{d_m z^X} \end{bmatrix} \end{bmatrix}.$$

The forward kinematics problem is formulated as the constrained nonlinear least-squares optimization problem

$$\min_{\rho \in \mathbb{R}^{3+6}, \Phi(\mathbf{q}^{ba})=\mathbf{0}} \mathbf{f}^T(\rho, \mathbf{v}_\ell) \mathbf{W}^{-1}(\rho) \mathbf{f}(\rho, \mathbf{v}_\ell), \quad (22)$$

where $\mathbf{W}(\rho)$ is defined in the same as in Section 3.1. The choice of objective function and the inclusion of $\mathbf{W}^{-1}(\rho)$ in the objective function is motivated by the same reasoning detailed in Section 3.1. The Levenberg–Marquardt algorithm is used to solve the optimization problem in (22) by iteratively solving for perturbations about $\bar{\rho}$ with the linearization and projection to the coordinates $\delta\hat{\rho}$ in (21), which leads to the unconstrained least-squares optimization problem

$$\min_{\delta\hat{\rho} \in \mathbb{R}^6} (\mathbf{f}(\bar{\rho}, \mathbf{0}) + \hat{\mathbf{J}}_\rho(\bar{\rho})\delta\hat{\rho})^T \mathbf{W}^{-1}(\bar{\rho}) (\mathbf{f}(\bar{\rho}, \mathbf{0}) + \hat{\mathbf{J}}_\rho(\bar{\rho})\delta\hat{\rho}). \quad (23)$$

Similarly to Section 3.1, the term $\mathbf{J}_v(\bar{\rho})\mathbf{v}_\ell$ in the linearization of (21) is not present in (11), as \mathbf{v}_ℓ is zero-mean unmeasurable noise that does not affect the objective function in (23) when averaged across multiple samples. The Levenberg–Marquardt algorithm solves (23) at the j th iteration with the pose estimate update

$$\delta\hat{\rho} = -(\hat{\mathbf{J}}_\rho^\top(\rho_j) \mathbf{W}^{-1}(\rho_j) \hat{\mathbf{J}}_\rho(\rho_j) + \eta \mathbf{1})^{-1} \hat{\mathbf{J}}_\rho^\top(\rho_j) \mathbf{W}^{-1}(\rho_j) \mathbf{f}(\rho_j, \mathbf{0}).$$

The end-effector position is then updated using

$$\mathbf{r}_{a,j+1}^{zo} = \mathbf{r}_{a,j}^{zo} + [\mathbf{1} \quad \mathbf{0}] \delta\hat{\rho}, \quad (24)$$

while the DCM describing the end-effector attitude is determined through the multiplicative update

$$\mathbf{C}_{b_{j+1}a}(\mathbf{q}^{b_{j+1}a}) = \exp(-\delta\psi_j^\times) \mathbf{C}_{b_ja}(\mathbf{q}^{b_ja}), \quad (25)$$

where $\delta\psi_j = [\mathbf{0} \quad \mathbf{1}] \delta\hat{\rho}$,

$$\exp(-\delta\psi_j^\times) = \mathbf{1} - \frac{\sin(\delta\psi_j)}{\delta\psi_j} \delta\psi_j^\times + \frac{1 - \cos(\delta\psi_j)}{\delta\psi_j^2} \delta\psi_j^\times \delta\psi_j^\times,$$

and $\delta\psi_j = \|\delta\psi_j\|_2$ [34]. The constrained attitude parameterization $\mathbf{q}^{b_{j+1}a}$ can then be extracted from $\mathbf{C}_{b_{j+1}a}(\mathbf{q}^{b_{j+1}a})$, ensuring that $\mathbf{q}^{b_{j+1}a}$ satisfies the attitude parameterization constraint. Examples of methods to perform this extraction for axis-angle parameters and quaternions are found in [44, pp. 21–27]. The covariance of the pose estimation error in the $\delta\hat{\rho}$ coordinates is calculated as [39, p. 84]

$$\hat{\mathbf{P}}_j = \mathbb{E}[\delta\hat{\rho}_j \delta\hat{\rho}_j^\top] = (\hat{\mathbf{J}}_\rho^\top(\rho_j) \mathbf{W}^{-1}(\rho_j) \hat{\mathbf{J}}_\rho(\rho_j))^{-1}.$$

To obtain the covariance of the pose estimation error in terms of the constrained attitude parameterization, the linear approximation $\rho_{\text{true}} - \rho_j \approx \Gamma(\mathbf{q}^{b_ja})\delta\hat{\rho}_j$ is determined from (19) to obtain

$$\begin{aligned} \mathbf{P}_j &= \mathbb{E}[(\rho_{\text{true}} - \rho_j)(\rho_{\text{true}} - \rho_j)^\top] \\ &\approx \mathbb{E}[\hat{\Gamma}(\mathbf{q}^{b_ja})\delta\hat{\rho}_j \delta\hat{\rho}_j^\top \hat{\Gamma}^\top(\mathbf{q}^{b_ja})] \\ &= \hat{\Gamma}(\mathbf{q}^{b_ja}) \mathbb{E}[\delta\hat{\rho}_j \delta\hat{\rho}_j^\top] \hat{\Gamma}^\top(\mathbf{q}^{b_ja}) \end{aligned} \quad (26)$$

$$= \hat{\Gamma}(\mathbf{q}^{b_ja}) \hat{\mathbf{P}}_j \hat{\Gamma}^\top(\mathbf{q}^{b_ja}), \quad (27)$$

where $\rho_{\text{true}} \in \mathbb{R}^n$ is the true end-effector pose with the constrained attitude parameterization.

4.2. Method 2: Cable-length loop-closure equations

The nonlinear cable-length loop-closure equations in (4) are linearized to obtain (12), whose terms are defined the same as in Section 3.2, except $\delta\rho^\top = [\delta\mathbf{r}_a^{zo^\top} \quad \delta\mathbf{q}^{ba^\top}]$ and the constrained attitude parameterization \mathbf{q}^{ba} replaces \mathbf{p}^{ba} in the expression for $\mathbf{J}_\rho(\rho)$ in (13). As in Section 4.1, the identity in (1) cannot be used to simplify (13). The attitude constraint equation $\Phi(\mathbf{q}^{ba}) = \mathbf{0}$ is linearized following the same procedure in Section 4.1, which results in $\mathbf{0} \approx \hat{\Xi}(\bar{\mathbf{q}}^{ba})\delta\rho$. Combining the linearized loop-closure equations and the attitude constraint, and then substituting in the coordinate transformation from (19) gives

$$\begin{bmatrix} \ell - \mathbf{g}(\bar{\rho}) - \mathbf{v}_\ell \\ \mathbf{0} \end{bmatrix} = \begin{bmatrix} \mathbf{H}(\bar{\rho}) \\ \hat{\Xi}(\bar{\mathbf{q}}^{ba}) \end{bmatrix} \delta\rho = \begin{bmatrix} \mathbf{H}(\bar{\rho}) \hat{\Gamma}(\bar{\mathbf{q}}^{ba}) \\ \hat{\Xi}(\bar{\mathbf{q}}^{ba}) \hat{\Gamma}(\bar{\mathbf{q}}^{ba}) \end{bmatrix} \delta\hat{\rho}. \quad (28)$$

The property $\hat{\Xi}(\bar{\mathbf{q}}^{ba}) \hat{\Gamma}(\bar{\mathbf{q}}^{ba}) = \mathbf{0}$ is used to simplify (28) to

$$\ell - \mathbf{g}(\bar{\rho}) - \mathbf{v}_\ell = \hat{\mathbf{H}}(\bar{\rho}) \delta\hat{\rho}, \quad (29)$$

where $\hat{\mathbf{H}}(\bar{\rho}) = \mathbf{H}(\bar{\rho}) \hat{\Gamma}(\bar{\mathbf{q}}^{ba})$. The constrained attitude parameterization identity in (2) is used to obtain the analytic expression

$$\hat{\mathbf{H}}(\rho) = \begin{bmatrix} \frac{\mathbf{r}_a^{d_1 w_1}(\rho)}{\|\mathbf{r}_a^{d_1 w_1}(\rho)\|_2} \begin{bmatrix} \mathbf{1} & -\mathbf{C}_{ba}^\top(\mathbf{q}^{ba}) \mathbf{r}_b^{d_1 z^\times} \end{bmatrix} \\ \vdots \\ \frac{\mathbf{r}_a^{d_m w_m}(\rho)}{\|\mathbf{r}_a^{d_m w_m}(\rho)\|_2} \begin{bmatrix} \mathbf{1} & -\mathbf{C}_{ba}^\top(\mathbf{q}^{ba}) \mathbf{r}_b^{d_m z^\times} \end{bmatrix} \end{bmatrix}.$$

Similarly to Section 3.2, it is interesting to note that taking the time derivative of (29) and assuming that the measurement noise is negligible results in $\dot{\ell} \approx \hat{\mathbf{H}}(\bar{\rho})\delta\dot{\hat{\rho}}$. This demonstrates that $\hat{\mathbf{H}}(\bar{\rho})$ is the Jacobian mapping from end-effector pose rates to cable length rates evaluated at $\bar{\rho}$.

The forward kinematics problem is formulated as the constrained nonlinear least-squares optimization problem

$$\min_{\rho \in \mathbb{R}^{3+n}, \Phi(\mathbf{q}^{ba})=\mathbf{0}} (\ell - \mathbf{g}(\rho))^\top \mathbf{V}^{-1} (\ell - \mathbf{g}(\rho)). \quad (30)$$

The choice of objective function and the inclusion of \mathbf{V}^{-1} in the objective function is motivated by the same reasoning detailed in Section 3.2. The Levenberg–Marquardt algorithm is used to solve the optimization problem in (30) by iteratively solving for perturbations about $\bar{\rho}$ with the linearization and projection to the coordinates $\delta\hat{\rho}$ in (29), which leads to the unconstrained least-squares optimization problem

$$\min_{\delta\hat{\rho} \in \mathbb{R}^6} (\ell - \mathbf{g}(\bar{\rho}) + \hat{\mathbf{H}}(\bar{\rho})\delta\hat{\rho})^\top \mathbf{V}^{-1} (\ell - \mathbf{g}(\bar{\rho}) + \hat{\mathbf{H}}(\bar{\rho})\delta\hat{\rho}). \quad (31)$$

The Levenberg–Marquardt algorithm solves (31) at the j th iteration with the pose estimate update

$$\delta\hat{\rho} = (\hat{\mathbf{H}}^T(\rho_j)\mathbf{V}^{-1}\hat{\mathbf{H}}(\rho_j) + \eta\mathbf{1})^{-1}\hat{\mathbf{H}}^T(\rho_j)\mathbf{V}^{-1}(\mathbf{e} - \mathbf{g}(\rho_j)).$$

The position estimate is updated additively according to (24) and the attitude estimate is updated multiplicatively through the DCM using (25). The covariance of the pose estimation error in the $\delta\hat{\rho}$ coordinates can be found directly as [39, p. 84]

$$\hat{\mathbf{P}}_j = \mathbb{E}[\delta\hat{\rho}_j\delta\hat{\rho}_j^T] = (\hat{\mathbf{H}}^T(\rho_j)\mathbf{V}^{-1}\hat{\mathbf{H}}(\rho_j))^{-1}. \quad (32)$$

Similar to Method 1, this can be mapped to a covariance of the pose estimation error in the ρ coordinates using the transformation in (27) with $\hat{\mathbf{P}}_j$ computed in (32).

Algorithm 2 Forward Kinematics Method 1 & Method 2 with Constrained Attitude Parameterizations

```

1: Measure  $\mathbf{e}$ 
2: Set  $\rho_0$ ,  $j = 0$ 
3: while Stopping criteria not met do
4:   if Method 1 then
5:     Calculate  $\mathbf{f} = \mathbf{f}(\rho_j, \mathbf{0})$ ,  $\hat{\mathbf{J}}_\rho = \hat{\mathbf{J}}_\rho(\rho_j)$ ,  $\mathbf{W} = \mathbf{W}(\rho_j)$ 
6:      $\delta\hat{\rho} = -(\hat{\mathbf{J}}_\rho^T\mathbf{W}^{-1}\hat{\mathbf{J}}_\rho + \eta\mathbf{1})^{-1}\hat{\mathbf{J}}_\rho^T\mathbf{W}^{-1}\mathbf{f}$ 
7:   else if Method 2 then
8:     Calculate  $\mathbf{g} = \mathbf{g}(\rho_j)$ ,  $\hat{\mathbf{H}} = \hat{\mathbf{H}}(\rho_j)$ 
9:      $\delta\hat{\rho} = (\hat{\mathbf{H}}^T\mathbf{V}^{-1}\hat{\mathbf{H}} + \eta\mathbf{1})^{-1}\hat{\mathbf{H}}^T\mathbf{V}^{-1}(\mathbf{e} - \mathbf{g})$ 
10:  end if
11:   $\mathbf{r}_{a,j+1}^{zo} = \mathbf{r}_{a,j}^{zo} + [\begin{matrix} \mathbf{1} & \mathbf{0} \end{matrix}] \delta\hat{\rho}$ 
12:   $\delta\psi_j = [\begin{matrix} \mathbf{0} & \mathbf{1} \end{matrix}] \delta\hat{\rho}$ 
13:   $\mathbf{C}_{b_{j+1}a} = \exp(-\delta\psi_j^\times) \mathbf{C}_{b_ja}(\mathbf{q}^{b_ja})$ 
14:  Extract  $\mathbf{q}^{b_{j+1}a}$  from  $\mathbf{C}_{b_{j+1}a}$ 
15:   $\rho_{j+1}^T = [\mathbf{r}_{a,j+1}^{zoT} \quad \mathbf{q}^{b_{j+1}aT}]$ 
16:  Set  $j = j + 1$ 
17: end while
18: if Method 1 then
19:   Calculate  $\mathbf{P}_j = \hat{\mathbf{F}}(\mathbf{q}^{b_ja}) (\hat{\mathbf{J}}_\rho^T\mathbf{W}^{-1}\hat{\mathbf{J}}_\rho)^{-1} \hat{\mathbf{F}}^T(\mathbf{q}^{b_ja})$ 
20: else if Method 2 then
21:   Calculate  $\mathbf{P}_j = \hat{\mathbf{F}}(\mathbf{q}^{b_ja}) (\hat{\mathbf{H}}^T\mathbf{V}^{-1}\hat{\mathbf{H}})^{-1} \hat{\mathbf{F}}^T(\mathbf{q}^{b_ja})$ 
22: end if
23: Return  $\rho_j$  and  $\mathbf{P}_j$ 

```

4.3. Discussion

Algorithm 2 contains a summary of the proposed forward kinematics methods for constrained attitude parameterizations. The main difference between the forward kinematics formulations for constrained and unconstrained attitude parameterizations lies in the calculations used to find $\delta\rho$. The update $\delta\rho$ is not performed in an additive fashion, as this would violate the attitude parameterization constraint, which is why the update is done with an unconstrained differential rotation vector and then converted back to the constrained attitude parameterization. This formulation requires an additional mapping to obtain an approximation of the estimation error covariance in the constrained attitude parameters. As with the unconstrained formulation, multiple solutions to the CDPR forward kinematics are possible and although they are not explicitly accounted for, warm start strategies can be used to limit this issue in practice.

Method 1 and Method 2 proposed in this section have all of the same benefits of Method 1 and Method 2 that are discussed in Section 3.3 with regards to the usefulness of having a pose estimation error covariance matrix available. The benefit provided by the algorithms presented in this section is that this pose estimation error covariance can be computed directly with the constrained attitude parameterization of choice.

5. Numerical simulation results

Numerical inverse kinematics simulations are performed with an eight-cable, six-DOF CDPR, similar to Fig. 2. The geometry of the CDPR is described in Table 1 through the position of the winch/cable anchor points, $\mathbf{r}_a^{w_i0}$, and the end-effector attachment points, $\mathbf{r}_b^{d_i z}$. A crossed-cable configuration similar to the IPAnema 2 setup described in [42, p. 319] is used, which results in a relatively large wrench-feasible translational and rotational workspace while avoiding cable collisions. A fixed time step of $t = 0.001$ s is used along with the kinematic equations $\mathbf{C}_{b_k a} = \exp(-t\omega_{b_k a}^\times) \mathbf{C}_{b_{k-1} a}$, $\mathbf{r}_{a_k}^{zo} = \mathbf{r}_{a_{k-1}}^{zo} + t\mathbf{v}_{a_{k-1}}^{zo/a}$, and $\mathbf{v}_{a_k}^{zo/a} = \mathbf{v}_{a_{k-1}}^{zo/a} + t\mathbf{a}_{a_{k-1}}^{zo/a/a}$ to simulate the

Table 1
Geometry of the CDPR in the simulations of Section 5.

Cable Number	Winch/Anchor point position, $\mathbf{r}_a^{w_0\text{T}}$ (cm)	End-effector attachment Point, $\mathbf{r}_b^{d,z\text{T}}$ (cm)
1	[71.5 38 93]	[3.75 7.5 -3.75]
2	[71.5 -38 93]	[3.75 -7.5 -3.75]
3	[-71.5 -38 93]	[-3.75 -7.5 -3.75]
4	[-71.5 38 93]	[-3.75 7.5 -3.75]
5	[71.5 38 0]	[3.75 3.75 3.75]
6	[71.5 -38 0]	[3.75 -3.75 3.75]
7	[-71.5 -38 0]	[-3.75 -3.75 3.75]
8	[-71.5 38 0]	[-3.75 3.75 3.75]

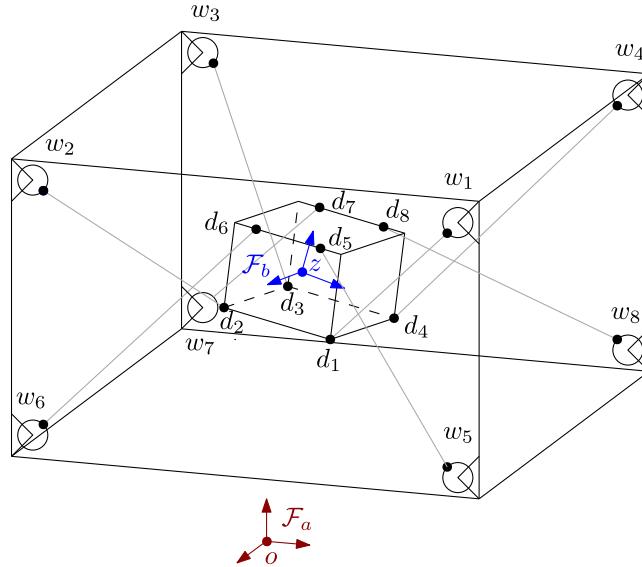


Fig. 2. Schematic of the CDPR in the simulations of Section 5.

end-effector trajectory, where the acceleration of the end-effector is chosen as $\mathbf{a}_{ak}^{zo/a/a^T} = -0.15 [\cos(kt) \quad \cos(kt) \quad \sin(kt)]$ m/s², $\omega_{bk}^{ba^T} = -13.5 [\cos(1.5kt) \quad 2\cos(1.5kt) \quad \cos(1.5kt)]$ rad/s is the angular velocity of the end-effector, $\mathbf{v}_{ak}^{zo/a}$ is the velocity of the end-effector, and the subscript $k \in \mathbb{N}$ denotes the quantities at the k th discrete time step. The initial conditions for the end-effector are chosen as $\mathbf{C}_{b0a} = \mathbf{1}$, $\mathbf{r}_{a0}^{zo^T} = [0.15 \quad 0.15 \quad 0.465]$ m, and $\mathbf{v}_{a0}^{zo/a^T} = [0.0005 \quad 0.0005 \quad 0.15]$ m/s. This results in the end-effector trajectory shown in Fig. 3. An inverse kinematics code is used to compute the cable lengths associated with the end-effector pose at each step in time. Zero-mean Gaussian noise $v_{\ell_i} \sim \mathcal{N}(0, \sigma_{\ell_i}^2)$, with $\sigma_{\ell_i} = 1$ mm, $i = 1, \dots, 8$, is then added to each cable length at each time step to simulate noisy cable length measurements.

The proposed forward kinematics methods outlined in Sections 3 and 4 are implemented in simulation. Both Method 1 and Method 2 are considered, where a 3-2-1 Euler-angle sequence is used as an unconstrained attitude parameterization, while the quaternion and the DCM are used for the constrained case. At the initial time step, all elements in ρ are initialized as $\mathbf{0}$. In practice, it is suggested to initialize the iterative Levenberg–Marquardt algorithm at a given time step with the solution from the previous time step, based on the assumption that the end-effector pose will not change drastically over a single time step. For the purposes of the simulations in the section, all iterations are initialized as $\mathbf{0}$, which serves as a worst-case estimate on the number of iterations required and provides consistent characterizations of performance across time steps.

In all cases, the damping coefficient in the Levenberg–Marquardt algorithm is set to $\eta = 0.001$. Two separate stopping criteria are considered: a tolerance on the magnitude of the updated pose estimate, chosen as $\|\delta\rho\|_2 < 10^{-9}$, and a maximum number of iterations set to 3. Simulation runs are performed with each individual stopping criteria to quantify the difference between the proposed methods in terms of error convergence and number of iterations.

5.1. Euler-angle sequence results

Method 1 and Method 2 are implemented with a 3-2-1 Euler-angle sequence to parameterize the end-effector attitude. Fig. 4 displays the pose estimation error with Method 2 for a single simulation run, where the stopping criterion is $\|\delta\rho\|_2 < 10^{-9}$. The estimation errors $\rho_k - \rho_{k,\text{true}} = [\tilde{r}_{1k} \quad \tilde{r}_{2k} \quad \tilde{r}_{3k} \quad \tilde{\theta}_{1k} \quad \tilde{\theta}_{2k} \quad \tilde{\theta}_{3k}]^T$ are computed at the k th time step and are shown in Fig. 4. The

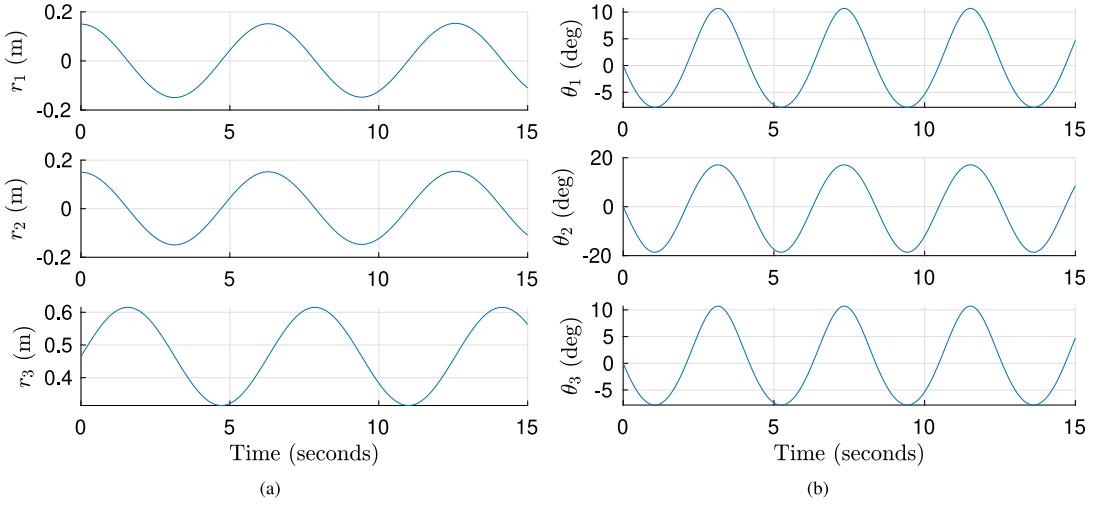


Fig. 3. The CDPR end-effector (a) position and (b) attitude simulated in Section 5. Note that the DCM is used in simulation and a 3-2-1 Euler-angle sequence is only used to conveniently visualize the end-effector attitude.

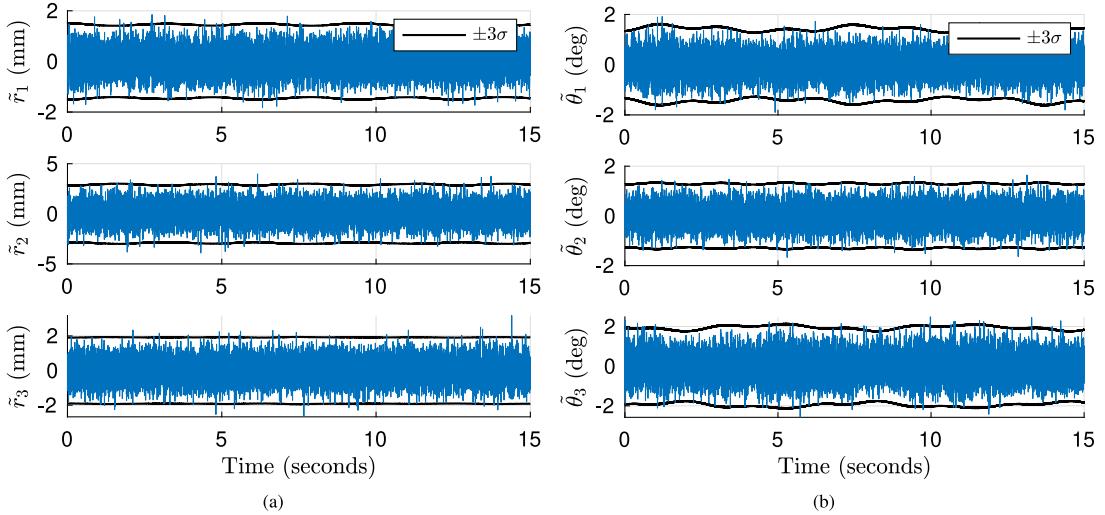


Fig. 4. The (a) position and (b) attitude errors for a single Euler-angle-based simulation in Section 5.1 using Method 2 with the stopping criterion $\|\delta\rho\|_2 < 10^{-9}$. The $\pm 3\sigma$ computed using the covariance matrix are included and shown to accurately bound the error.

3σ bounds on the pose estimation error calculated by taking the square root of the diagonal entries of the covariance \mathbf{P}_j at each time step are included in these plots to illustrate the predicted confidence in the estimation error. The 3σ bounds in Fig. 4 appear to accurately reflect the pose estimation error, which is on the order of 1–2 mm in position and 1–2 deg in attitude. Results with Method 1 are not included, as they are indistinguishable from the results with Method 2.

In order to validate the accuracy of the computed covariance matrices \mathbf{P}_j relative to the true statistics on the pose estimation errors, a batch of 100 Monte-Carlo simulations with varying measurement noise samples are performed for the two forward kinematics algorithms. Specifically, the end-effector trajectory is identical across all of the Monte-Carlo simulations, while the measurement noise added to the cable length measurements is randomly sampled from a zero-mean Gaussian distribution with a standard deviation of 1 mm in each simulation run. Although the measurement noise added to each simulation run is drawn from the same distribution, the specific value of noise added at particular instances in time differs between simulation runs.

An estimator whose covariances accurately represent the estimation errors is considered consistent and performing in an optimal fashion. To perform a check of consistency, the normalized estimation error squared (NEES) test is used [39, pp. 301–304]. This test is performed by first computing the NEES at each discrete time step of the simulation as

$$\epsilon_k = (\rho_{\text{true},k} - \rho_{j,k})^T \mathbf{P}_{j,k}^{-1} (\rho_{\text{true},k} - \rho_{j,k}), \quad (33)$$

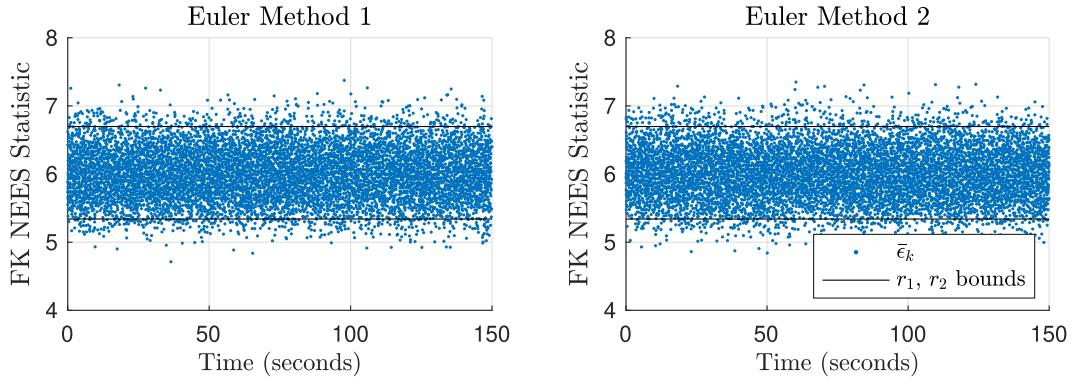


Fig. 5. Plots of the average NEES ($\bar{\epsilon}_k$) versus time from the Monte Carlo simulations performed in Section 5.1 with Method 1 and Method 2 for the Euler-angle-based forward kinematics algorithm and a stopping criterion $\|\delta\rho\|_2 < 10^{-9}$. The black horizontal lines indicate chi-square tail probability bounds associated with a confidence interval of 95%. The results show 94.72% of the average NEES lies within these bounds with Method 1 and 94.77% with Method 2.

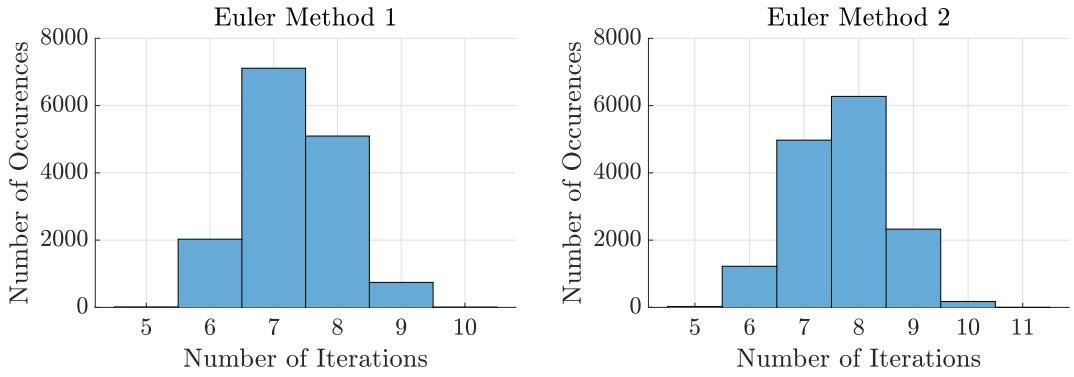


Fig. 6. Histogram of the number of iterations needed for Method 1 and Method 2 to converge to a tolerance of $\|\delta\rho\|_2 < 10^{-9}$ for the Euler-angle-based Monte-Carlo simulations in Section 5.1. The average number of iterations is 7.30 with Method 1 and 7.68 with Method 2.

where $\rho_{j,k}$ is the pose estimate and $\mathbf{P}_{j,k}$ is the computed covariance at the k th time step following j iterations of the Levenberg–Marquardt iterations, while $\rho_{\text{true},k}$ is the true pose at the k th time step. If the NEES is generated from a consistent estimator, then the NEES is to follow a chi-square distribution with degrees of freedom matching the dimension of the state at each time step. A chi-square hypothesis test is performed to assess the consistency of the forward kinematics algorithms. This involves evaluating the average NEES for each method across N Monte Carlo simulation runs as $\bar{\epsilon}_k = \frac{1}{N} \sum_{i=1}^N \epsilon_k^i$, where ϵ_k^i is the NEES of the i th simulation at the k th time step. If $\bar{\epsilon}_k$ lies within the tail probabilities associated with level α , then the estimator is declared consistent with level α . Specifically, this is assessed by computing the tail probability bounds $r_1 = \chi_{6N}^2(\alpha/2)/N$ and $r_2 = \chi_{6N}^2(1 - \alpha/2)/N$, where $\chi_i^2(\cdot)$ is the chi-square cumulative distribution function with i degrees of freedom [39, p. 302]. The $6N$ degrees of freedom in this problem are a result of the 6-dimensional state vector and the N Monte-Carlo simulation runs. The variable $0 < \alpha < 1$ determines the confidence level $1 - \alpha$ (e.g., $\alpha = 0.05$ indicates a 95% confidence interval).

A plot of $\bar{\epsilon}_k$ versus time alongside the computed 95% confidence interval bounds r_1 and r_2 is included in Fig. 5 for Method 1 and Method 2, where it is shown that 94.72% and 94.77% of the values of $\bar{\epsilon}_k$ lie within the bounds r_1 and r_2 with Method 1 and Method 2, respectively. This indicates that the covariance matrices computed with both methods are accurate and the estimator is consistent.

Histograms of the number of iterations required to meet the stopping criterion $\|\delta\rho\|_2 < 10^{-9}$ for Method 1 and Method 2 are presented in Fig. 6. Method 2 requires a larger number of iterations than Method 1 (average of 7.68 iterations with Method 2 compared to 7.30 with Method 1).

To further investigate the differences between Method 1 and Method 2, Fig. 7 presents the average root mean square error (RMSE) in the estimated pose over 100 Monte Carlo simulations, where only 3 iterations of the Levenberg–Marquardt are allowed. It is shown that Method 1 converges to a smaller estimation error compared to Method 2 under this constraint of very limited iterations.

5.2. Quaternion results

Method 1 and Method 2 are implemented with the use of the quaternion to parameterize the end-effector attitude. Fig. 8 displays the pose estimation error with Method 2 for a single simulation run, where the stopping criterion is $\|\delta\rho\|_2 < 10^{-9}$. The estimation

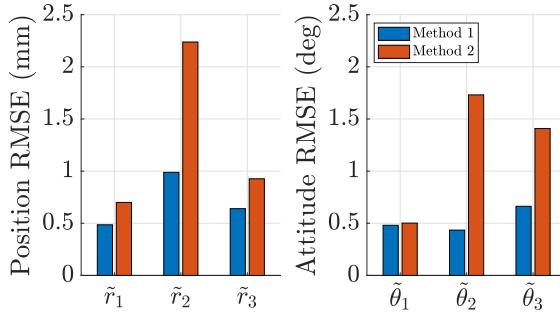


Fig. 7. Bar plots of the average RMSE with Method 1 and Method 2 using 3 iterations for the Euler-angle-based Monte-Carlo simulations in Section 5.1.

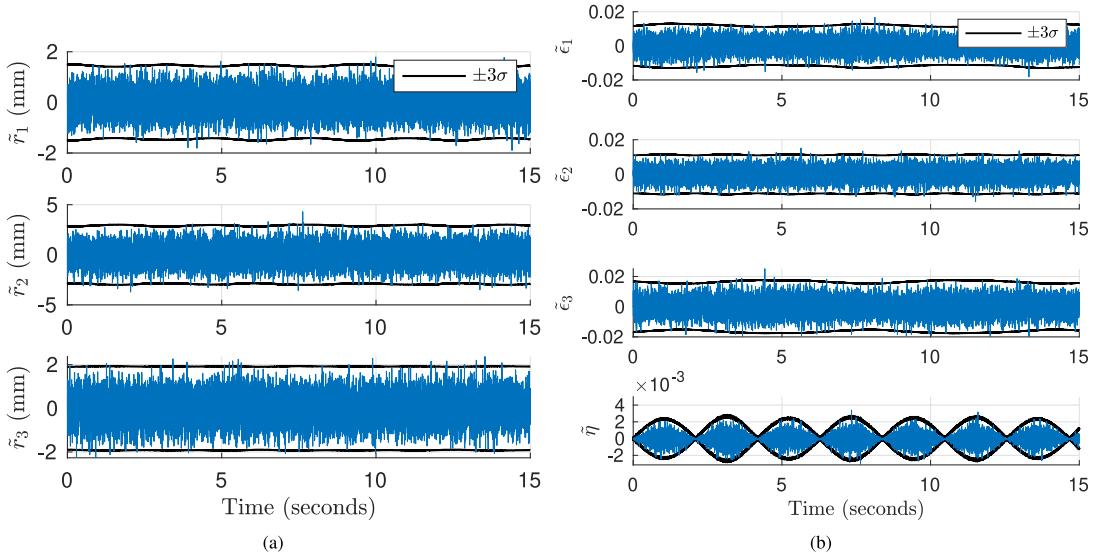


Fig. 8. The (a) position and (b) attitude errors for a single quaternion-based simulation in Section 5.2 using Method 2 with the stopping criterion $\|\delta\rho\|_2 < 10^{-9}$. The $\pm 3\sigma$ computed using the covariance matrix are included and shown to accurately bound the error.

errors $\rho_k - \rho_{k,\text{true}} = [\tilde{r}_{1k} \quad \tilde{r}_{2k} \quad \tilde{r}_{3k} \quad \tilde{\epsilon}_{1k} \quad \tilde{\epsilon}_{2k} \quad \tilde{\epsilon}_{3k} \quad \tilde{\eta}_k]^T$ are computed at the k th time step and are shown in Fig. 8. The 3σ bounds on the pose estimation error calculated using the covariance matrix \mathbf{P}_j at each time step and are included in these plots to illustrate the predicted confidence in the estimation error. Similarly to the Euler-angle case, a batch of 100 Monte-Carlo simulations with varying measurement noise samples are performed for the two forward kinematics algorithms. In order to validate the accuracy of the computed covariance matrices \mathbf{P}_j relative to the true statistics on the pose estimation errors, the NEES test is used in a similar fashion to Section 5.1. Due to the constrained nature of the quaternion, the NEES computation is slightly different and is given by

$$\epsilon_k = (\rho_{\text{true},k} - \rho_{j,k})^T \hat{\mathbf{S}}_{j,k} \hat{\mathbf{P}}_{j,k}^{-1} \hat{\mathbf{S}}_{j,k} (\rho_{\text{true},k} - \rho_{j,k}), \quad (34)$$

where $\rho_{j,k} \in \mathbb{R}^7$ is the pose estimate and $\hat{\mathbf{P}}_{j,k} \in \mathbb{R}^{6 \times 6}$ is the computed covariance in terms of the $\hat{\rho}$ coordinates at the k th time step following j iterations of the Levenberg–Marquardt algorithm, $\rho_{\text{true},k} \in \mathbb{R}^7$ is the true pose at the k th time step, $\hat{\mathbf{S}}_{j,k} = \text{diag}\{\mathbf{1}, \mathbf{S}(\mathbf{q}^{ba})\}$, and $\mathbf{S}(\mathbf{q}^{ba})$ satisfies $\omega_b^{ba} = \mathbf{S}(\mathbf{q}^{ba})\dot{\mathbf{q}}^{ba}$. This difference between the NEES computation in (33) and (34) is due to the fact that the full pose covariance with the quaternion, $\mathbf{P}_{j,k} \in \mathbb{R}^{7 \times 7}$, only has rank 6 and is therefore not invertible and not applicable for use with the NEES computation in (33). The NEES computation in (34) makes use of the approximation $\delta\hat{\rho} \approx \hat{\mathbf{S}}(\bar{\mathbf{q}}^{ba})\delta\rho$, which is derived as a inverse relation to the approximation in (19) using the fact that $\hat{\mathbf{S}}(\bar{\mathbf{q}}^{ba})\hat{\mathbf{f}}(\bar{\mathbf{q}}^{ba}) = \mathbf{1}$. With this in mind, the NEES in (34) is approximately equal to $(\hat{\rho}_{\text{true},k} - \hat{\rho}_{j,k})^T \hat{\mathbf{P}}_{j,k}^{-1} (\hat{\rho}_{\text{true},k} - \hat{\rho}_{j,k})$, where the error is computed in terms of the $\hat{\rho}$ coordinates. Other than this slight difference in NEES computation, the NEES test proceeds in the same manner as in Section 5.1. It is worth noting that the number of degrees of freedom in the chi-square hypothesis test remains $6N$ even though the quaternion-based pose has 7 states. This is because regardless of the attitude parameterization used, there are only 6 independent degrees of freedom in the pose. A plot of $\tilde{\epsilon}_k$ versus time alongside the computed bounds r_1 and r_2 is included in Fig. 9 for Method 1 and Method 2, where it is shown that 94.82% and 95.02% of the values of $\tilde{\epsilon}_k$ lie within the bounds r_1 and r_2 with Method 1 and Method 2, respectively. This matches well with the chosen 95% confidence interval and indicates that the covariance matrices computed with both methods are accurate and the estimator is consistent.

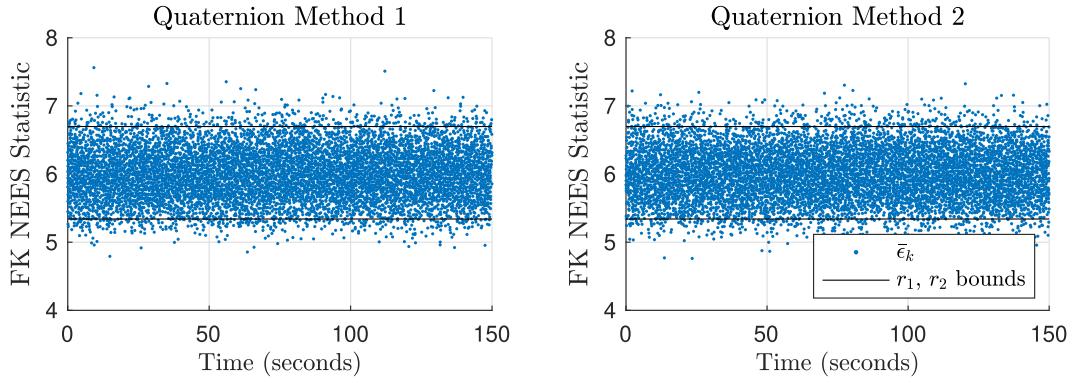


Fig. 9. Plots of the average NEES ($\bar{\epsilon}_k$) versus time from the Monte Carlo simulations performed in Section 5.2 with Method 1 and Method 2 for the quaternion-based forward kinematics algorithm and stopping criterion $\|\delta\rho\|_2 < 10^{-9}$. The black horizontal lines indicate chi-square tail probability bounds associated with a confidence interval of 95%. The results show 94.82% of the average NEES lies within these bounds with Method 1 and 95.02% with Method 2.

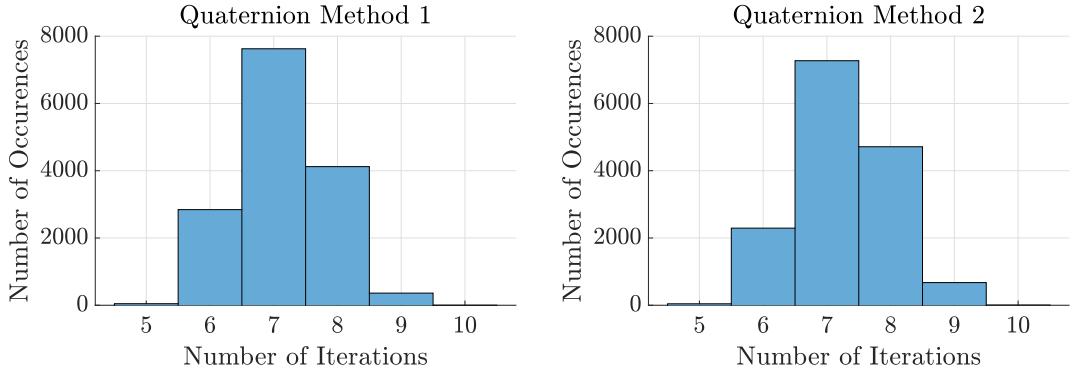


Fig. 10. Histogram of the number of iterations needed for Method 1 and Method 2 to converge to a tolerance of $\|\delta\rho\|_2 < 10^{-9}$ for the quaternion-based Monte-Carlo simulations in Section 5.2. The average number of iterations is 7.13 with Method 1 and 7.25 with Method 2.

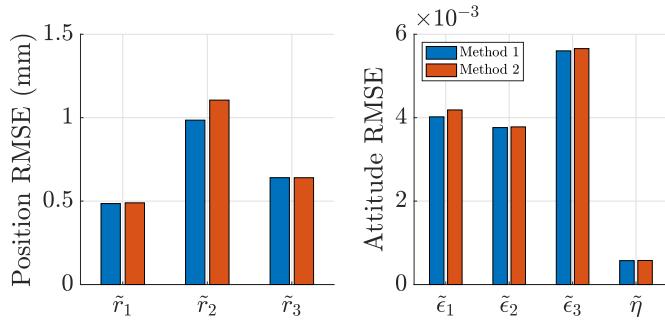


Fig. 11. Bar plots of the average RMSE with Method 1 and Method 2 using 3 iterations for the quaternion-based Monte-Carlo simulations in Section 5.2.

Histograms of the number of iterations performed by forward kinematics Methods 1 and 2 with a stopping criterion of $\|\delta\rho\|_2 < 10^{-9}$ are included in Fig. 10. As with the Euler-angle sequence simulations, Method 2 uses a larger number of iterations (a mean of 7.25 iterations) to meet the same stopping criterion compared to Method 1 (a mean of 7.13 iterations).

The average RMSE across Monte Carlo simulations with a stopping criterion of 3 iterations is shown in Fig. 11. As with the Euler-angle-based approach, Method 2 results in a larger RMSE compared to Method 1.

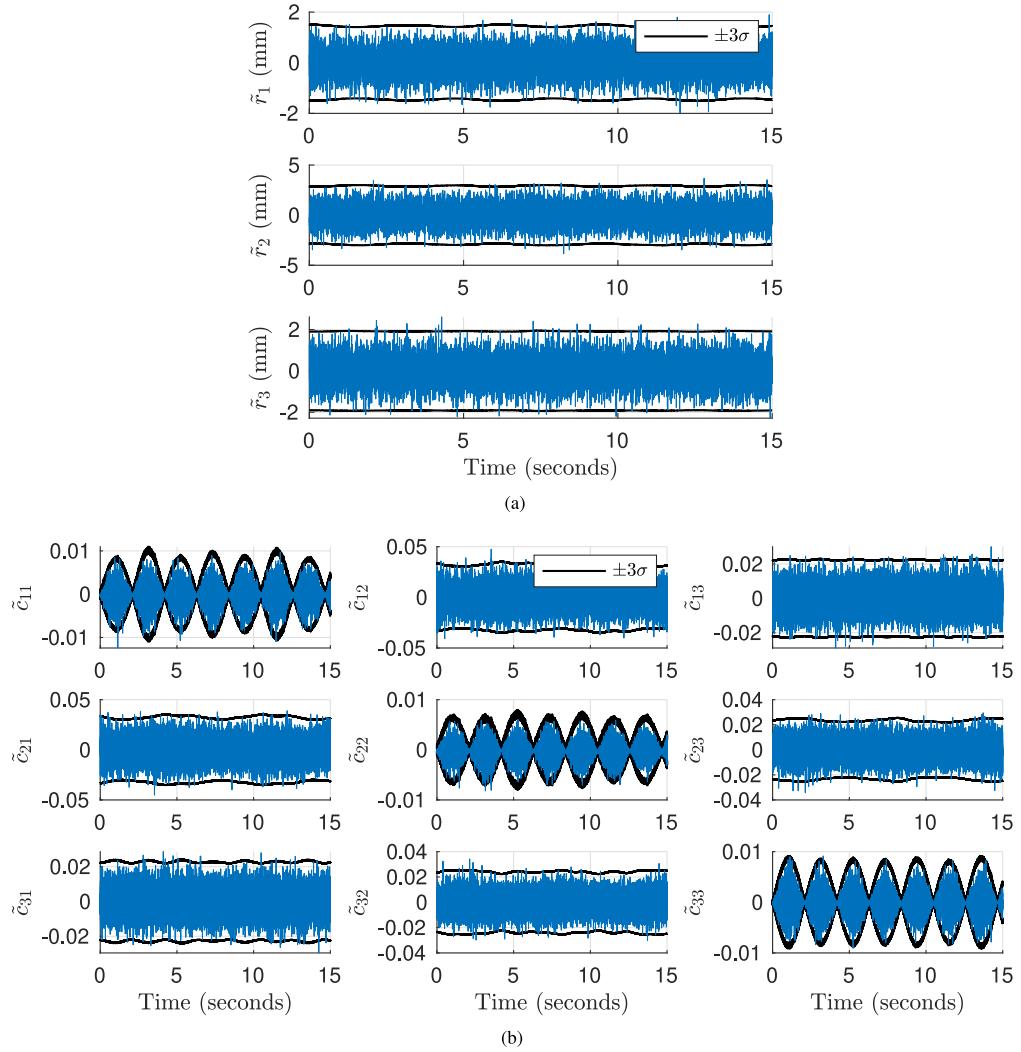


Fig. 12. The (a) position and (b) attitude errors for a single DCM-based simulation in Section 5.3 using Method 2 with the stopping criterion $\|\delta\rho\|_2 < 10^{-9}$, where \tilde{c}_{ij} denotes the estimation error in the i th row and j th column of the DCM. The $\pm 3\sigma$ computed using the covariance matrix are included and shown to accurately bound the error.

5.3. Direction cosine matrix results

Method 1 and Method 2 are also implemented with the DCM to parameterize the end-effector attitude. Fig. 12 includes the pose estimation error with Method 2 for a single simulation run, where the stopping criterion is $\|\delta\rho\|_2 < 10^{-9}$. The estimation errors

$$\rho_k - \rho_{k,\text{true}} = [\tilde{r}_{1k} \quad \tilde{r}_{2k} \quad \tilde{r}_{3k} \quad \tilde{\epsilon}_{1k} \quad \tilde{c}_{11k} \quad \tilde{c}_{21k} \quad \tilde{c}_{31k} \quad \tilde{c}_{12k} \quad \tilde{c}_{22k} \quad \tilde{c}_{32k} \quad \tilde{c}_{13k} \quad \tilde{c}_{23k} \quad \tilde{c}_{33k}]^T$$

are computed at the k th time step and are shown in Fig. 12, where \tilde{c}_{ijk} represents the error in the entry of the DCM's i th row and j th column at the k th time step. The 3σ bounds on the pose estimation error calculated using the covariance matrix \mathbf{P}_j at each time step and are included in these plots. A batch of 100 Monte-Carlo simulations with varying measurement noise samples are performed for the two forward kinematics algorithms. The accuracy of the computed covariance matrices \mathbf{P}_j relative to the true statistics on the pose estimation errors is validated through the NEES test, as done in Section 5.2. A plot of the average NEES, $\bar{\epsilon}_k$, versus time alongside the computed bounds r_1 and r_2 is included in Fig. 13 for Method 1 and Method 2, where it is shown that 94.74% and 94.43% of the values of $\bar{\epsilon}_k$ lie within the bounds r_1 and r_2 with Method 1 and Method 2, respectively. This matches well with the chosen 95% confidence interval and indicates that the covariance matrices computed with both methods are accurate and the estimator is consistent.

The number of iterations of the Levenberg–Marquardt algorithm with Methods 1 and 2 and a stopping criterion of $\|\delta\rho\|_2 < 10^{-9}$ are reported in Fig. 14. Method 2 uses a larger number of iterations (a mean of 7.49 iterations) to meet the same stopping criterion compared to Method 1 (a mean of 7.37 iterations).

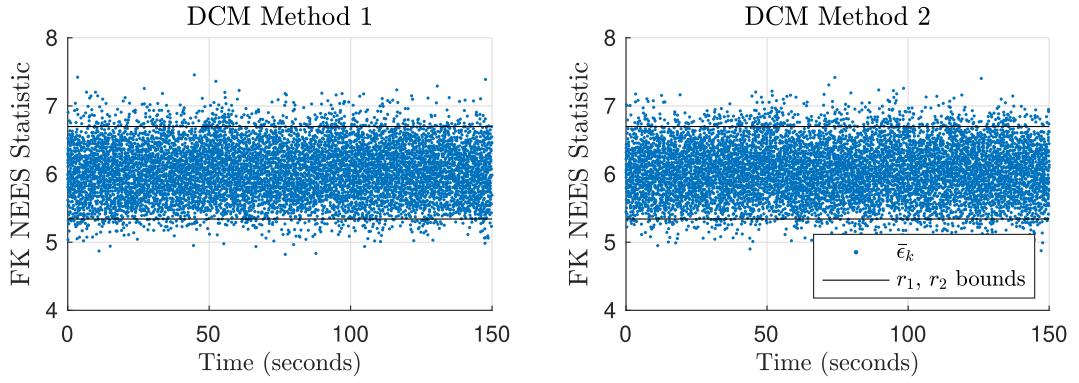


Fig. 13. Plots of the average NEES ($\bar{\epsilon}_k$) versus time from the Monte Carlo simulations performed in Section 5.3 with Method 1 and Method 2 for the DCM-based forward kinematics algorithm and the stopping criterion $\|\delta\rho\|_2 < 10^{-9}$. The black horizontal lines indicate chi-square tail probability bounds associated with a confidence interval of 95%. The results show 94.74% of the average NEES lies within these bounds with Method 1 and 94.43% with Method 2.

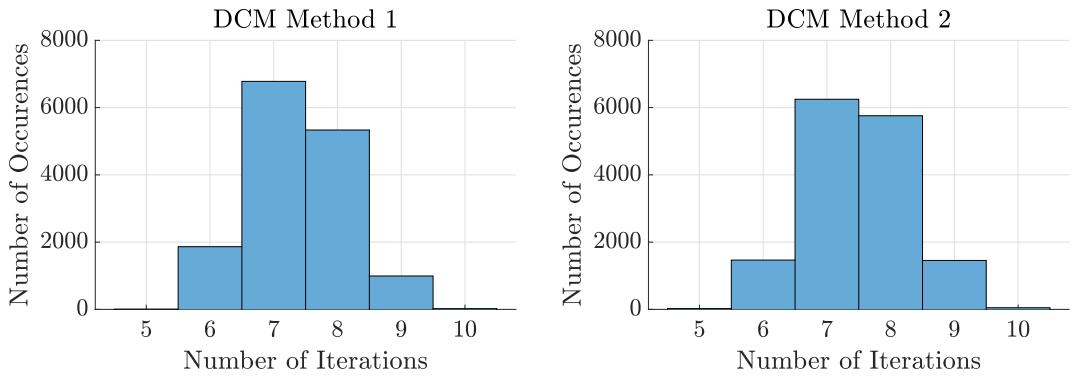


Fig. 14. Histogram of the number of iterations needed for Method 1 and Method 2 to converge to a tolerance of $\|\delta\rho\|_2 < 10^{-9}$ for the DCM-based Monte-Carlo simulations in Section 5.3. The average number of iterations is 7.37 with Method 1 and 7.49 with Method 2.

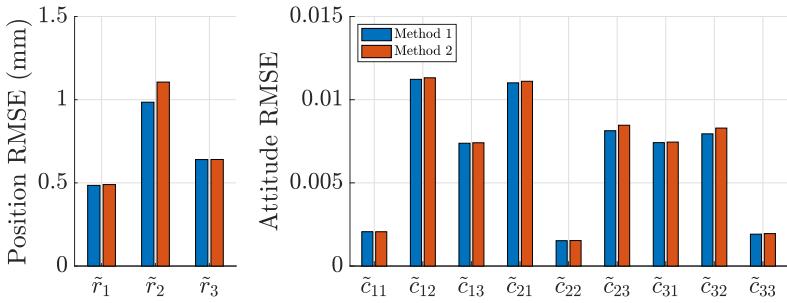


Fig. 15. Bar plots of the average RMSE with Method 1 and Method 2 using 3 iterations for the DCM-based Monte-Carlo simulations in Section 5.3, where \tilde{c}_{ij} denotes the estimation error in the i th row and j th column of the DCM.

The average RMSE across Monte Carlo simulations with a stopping criterion of 3 iterations is shown in Fig. 15, where Method 2 results in a larger RMSE compared to Method 1.

5.4. Discussion

The numerical results presented in this section with the proposed forward kinematics demonstrate that both Method 1 and Method 2 provide accurate and consistent pose estimates across different attitude parameterizations of the end-effector pose. Differences between Method 1 and Method 2 are seen in their convergence properties as part of the iterative Levenberg–Marquardt algorithm. On average, Method 2 (cable-length loop-closure equations) requires more iterations to achieve the same accuracy as Method 1 (cable-length-squared loop-closure equations) and achieves a less accurate pose estimate when limiting both methods

to the same number of iterations. This difference is largest in the case where an Euler-angle-sequence attitude parameterization is used. The reason for this difference is not entirely clear, but provides useful information that can help with the choice of a forward kinematics algorithm, particularly when quick convergence is required. It should be noted that this comparison between Method 1 and Method 2 is limited to the specific CDPR parameters and simulations performed in this work and further investigation is required to fully characterize their differences.

6. Conclusion

In this paper, iterative forward kinematics formulations were developed for use with unconstrained and constrained attitude parameterizations that explicitly compute covariance bounds on the pose estimation error. The proposed methods make use of attitude identities that result in analytical expressions for the terms in the iterative algorithm, including the covariance matrix. The numerical results demonstrated using the NEES test that the proposed methods can accurately assess the covariance on the pose estimation error, which emphasizes the practicality of the proposed approaches. This was confirmed through implementation with Euler-angle-sequence, quaternion, and DCM parameterizations of the end-effector attitude. The simulation results also indicated that although both forms of the proposed forward kinematics algorithm produce accurate pose estimates, the method that makes use of cable-length-squared loop-closure equations has better convergence properties compared to the method that uses cable-length loop-closure equations. The differences between these approaches will be investigated further in future work.

Additional future work on this topic will include further validation through experiments and an investigation into the effect of calibration errors and non-Gaussian measurement noise on the accuracy of the covariance bounds. Furthermore, the benefits of the proposed forward kinematics approaches will be assessed by incorporating them within a Kalman filtering framework that includes a kinematic or dynamic process model.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

The authors would like to thank Dr. Kirsten Strandjord for helpful discussions on the use of the NEES test. This work was supported in part by the University of Minnesota's Office of Undergraduate Research, United States.

References

- [1] Q. Chen, B. Zi, Z. Sun, Y. Li, Q. Xu, Design and development of a new cable-driven parallel robot for waist rehabilitation, *IEEE/ASME Trans. Mechatronics* 24 (4) (2019) 1497–1507.
- [2] E. Barnett, C. Gosselin, Large-scale 3D printing with a cable-suspended robot, *Addit. Manuf.* 7 (2015) 27–44.
- [3] P. Miermeister, M. Lächele, R. Boss, C. Masone, C. Schenk, J. Tesch, M. Kerger, H. Teufel, A. Pott, H.H. Bülfhoff, The CableRobot simulator large scale motion platform based on cable robot technology, in: *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, Daejeon, South Korea, 2016, pp. 3024–3029.
- [4] H. Sun, X. Tang, Z. Cui, S. Hou, Dynamic response of spatial flexible structures subjected to controllable force based on cable-driven parallel robots, *IEEE/ASME Trans. Mechatronics* 25 (6) (2020) 2801–2811.
- [5] A. Pott, An algorithm for real-time forward kinematics of cable-driven parallel robots, in: J. Lenarcic, M.M. Stanisic (Eds.), *Advances in Robot Kinematics: Motion in Man and Machine*, Springer, Dordrecht, Netherlands, 2010, pp. 529–538.
- [6] A. Pott, V. Schmidt, On the forward kinematics of cable-driven parallel robots, in: *IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS*, Hamburg, Germany, 2015, pp. 3182–3187.
- [7] J.C. Santos, M. Gouttefarde, A real-time capable forward kinematics algorithm for cable-driven parallel robots considering pulley kinematics, in: *Advances in Robot Kinematics 2020*, in: Springer Proceedings in Advanced Robotics, vol. 15, Springer, Cham, Switzerland, 2021, pp. 199–208.
- [8] A. Berti, J.-P. Merlet, M. Carricato, Solving the direct geometrico-static problem of underconstrained cable-driven parallel robots by interval analysis, *Int. J. Robot. Res.* 35 (6) (2016) 239–723.
- [9] C. Sancak, M. Itik, Out-of-plane vibration suppression and position control of a planar cable-driven robot, *IEEE/ASME Trans. Mechatronics* 27 (3) (2022) 1311–1320.
- [10] J.-P. Merlet, Some properties of the Irvine cable model and their use for the kinematic analysis of cable-driven parallel robots, *Mech. Mach. Theory* 135 (2019) 271–280.
- [11] Z. Zhang, G. Xie, Z. Shao, C. Gosselin, Kinematic calibration of cable-driven parallel robots considering the pulley kinematics, *Mech. Mach. Theory* 169 (2022) 104648.
- [12] R.J. Caverly, J.R. Forbes, State estimator design for a single degree of freedom cable-actuated system, *J. Franklin Inst. B* 353 (18) (2016) 4845–4869.
- [13] M.H. Korayem, M. Yousefzadeh, B. Beyranvand, Dynamics and control of a 6-dof cable-driven parallel robot with visco-elastic cables in presence of measurement noise, *J. Intell. Robot. Syst.* 88 (2017) 73–95.
- [14] M. Korayem, M. Yousefzadeh, S. Kian, Precise end-effector pose estimation in spatial cable-driven parallel robots with elastic cables using a data fusion method, *Measurement* 130 (2018) 177–190.

- [15] C.T. Schenk, Modelling and Control of a Cable-Driven Parallel Robot Methods for Vibration Reduction and Motion Quality Improvement (Ph.D. thesis), University of Stuttgart, Stuttgart, Germany, 2019.
- [16] R. Qi, M. Rushton, A. Khajepour, W.W. Melek, Decoupled modeling and model predictive control of a hybrid cable-driven robot (HCDR), *Robot. Auton. Syst.* 118 (2019) 1–12.
- [17] V.L. Nguyen, R.J. Caverly, Cable-driven parallel robot pose estimation using extended Kalman filtering with inertial payload measurements, *IEEE Robot. Autom. Lett.* 6 (2) (2021) 3615–3622.
- [18] R. Ramadour, F. Chaumette, J.-P. Merlet, Grasping objects with a cable-driven parallel robot designed for transfer operation by visual servoing, in: IEEE International Conference on Robotics and Automation, Hong Kong, China, 2014, pp. 4463–4468.
- [19] Z. Zake, F. Chaumette, N. Pedemonte, S. Caro, Vision-based control and stability analysis of a cable-driven parallel robot, *IEEE Robot. Autom. Lett.* 4 (2) (2019) 1029–1036.
- [20] T. Dallej, M. Gouttefarde, N. Andreff, R. Dahmouche, P. Martinet, Vision-based modeling and control of large-dimension cable-driven parallel robots, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, Vilamoura-Algarve, Portugal, 2012, pp. 1581–1586.
- [21] R. Chellal, L. Cuvillon, E. Laroche, A kinematic vision-based position control of a 6-DoF cable-driven parallel robot, in: A. Pott, T. Bruckmann (Eds.), *Cable-Driven Parallel Robots*, in: Mechanisms and Machine Science, vol. 32, Springer, Cham, Switzerland, 2015, pp. 213–225.
- [22] T. Dallej, M. Gouttefarde, N. Andreff, P.-E. Hervé, P. Martinet, Modeling and vision-based control of large-dimension cable-driven parallel robots using a multiple-camera setup, *Mechatronics* 61 (2019) 20–36.
- [23] A. Fortin-Côté, P. Cardou, A. Campeau-Lecours, Improving cable-driven parallel robot accuracy through angular position sensors, in: IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS, Daejeon, South Korea, 2016, pp. 4350–4355.
- [24] X. Garant, A. Campeau-Lecours, P. Cardou, C. Gosselin, Improving the forward kinematics of cable-driven parallel robots through cable angle sensors, in: C. Gosselin, P. Cardou, T. Bruckmann, A. Pott (Eds.), *Cable-Driven Parallel Robots*, in: Mechanisms and Machine Science, vol. 53, Springer, Cham, Switzerland, 2018, pp. 167–179.
- [25] C. Martin, M. Fabritius, J.T. Stoll, A. Pott, Accuracy improvement for CDPRs based on direct cable length measurement sensors, in: M. Gouttefarde, T. Bruckmann, A. Pott (Eds.), *Cable-Driven Parallel Robots: CableCon 2021*, in: Mechanisms and Machine Science, vol. 104, Springer, Cham, Switzerland, 2021, pp. 348–359.
- [26] V.L. Nguyen, R.J. Caverly, CDPR forward kinematics with error covariance bounds for unconstrained end-effector attitude parameterizations, in: M. Gouttefarde, T. Bruckmann, A. Pott (Eds.), *Cable-Driven Parallel Robots: CableCon 2021*, in: Mechanisms and Machine Science, vol. 104, Springer, Cham, Switzerland, 2021, pp. 37–49.
- [27] E.J. Lefferts, F.L. Markley, M.D. Shuster, Kalman filtering for spacecraft attitude estimation, *J. Guid. Control Dyn.* 5 (5) (1982) 417–429.
- [28] R. Mahony, T. Hamel, J.-M. Pflimlin, Nonlinear complementary filters on the special orthogonal group, *IEEE Trans. Automat. Control* 53 (5) (2008) 1203–1218.
- [29] T. Barfoot, J.R. Forbes, P.T. Furgale, Pose estimation using linearized rotations and quaternion algebra, *Acta Astronaut.* 68 (1–2) (2011) 101–112.
- [30] A. Barrau, S. Bonnabel, The invariant extended Kalman filter as a stable observer, *IEEE Trans. Automat. Control* 62 (4) (2016) 1797–1812.
- [31] B.E. Jackson, K. Tracy, Z. Manchester, Planning with attitude, *IEEE Robot. Autom. Lett.* 6 (3) (2021) 5658–5664.
- [32] T. Lee, M. Leok, N.H. McClamroch, Geometric tracking control of a quadrotor UAV on SE (3), in: IEEE Conference on Decision and Control, Atlanta, GA, 2010, pp. 5420–5425.
- [33] S.K. Cheah, R.J. Caverly, Passivity-based pose regulation and jacobian-based force distribution of a cable-driven parallel robot, in: American Control Conference, New Orleans, LA, 2021, pp. 124–129.
- [34] A.H. de Ruiter, J.R. Forbes, General identities for parameterizations of SO(3) with applications, *J. Appl. Mech.* 81 (7) (2014) 071007.
- [35] P.C. Hughes, *Spacecraft Attitude Dynamics*, Dover, Mineola, NY, 2004.
- [36] J.-P. Merlet, Solving the forward kinematics of a Gough-type parallel manipulator with interval analysis, *Int. J. Robot. Res.* 23 (3) (2004) 221–235.
- [37] J.-P. Merlet, An experimental investigation of extra measurements for solving the direct kinematics of cable-driven parallel robots, in: IEEE Int. Conf. Robot., 2018, pp. 6947–6952.
- [38] A. Jazwinski, *Stochastic Processes and Filtering Theory*, Dover, Mineola, NY, 2007.
- [39] J.L. Crassidis, J.L. Junkins, Optimal Estimation of Dynamic Systems, in: Applied Mathematics and Nonlinear Science, vol. 2, Chapman & Hall/CRC, Boca Raton, FL, 2004.
- [40] K. Madsen, H.B. Nielson, O. Tingleff, *Methods for Non-Linear Least Squares Problems*, second ed., IMM, Technical University of Denmark, Lyngby, Denmark, 2004.
- [41] H.P. Gavin, The Levenberg-Marquardt Algorithm for Nonlinear Least Squares Curve-Fitting Problems, Department of Civil and Environmental Engineering, Duke University, 2020, pp. 1–19.
- [42] A. Pott, *Cable-Driven Parallel Robots: Theory and Application*, in: Springer Tracts in Advanced Robotics, vol. 120, Springer International, Cham, Switzerland, 2018.
- [43] M. Fabritius, A. Pott, A forward kinematic code for cable-driven parallel robots considering cable sagging and pulleys, in: J. Lenarčič, B. Siciliano (Eds.), *Advances in Robot Kinematics 2020*, Springer, Cham, Switzerland, 2021, pp. 218–225.
- [44] A.H. de Ruiter, C. Damaren, J.R. Forbes, *Spacecraft Dynamics and Control: An Introduction*, Wiley & Sons, Chichester, UK, 2013.