Kinematics-Inertial Fusion for Localization of a 4-Cable Underactuated Suspended Robot Considering Cable Sag

Eren Allak[1], Rooholla Khorrambakht[2], Christian Brommer[1], and Stephan Weiss[1]

***Abstract*— Suspended Cable-Driven Parallel Robots (SCDPR) have intriguing capabilities on large scales but still have open challenges in precisely estimating the end-effector pose. The cables exhibit a downward curved shape, also known as *cable sag* which needs to be accounted for in the pose estimation. The catenary equations can accurately describe this phenomenon but are only accurate in equilibrium conditions. Thus, pose estimation for large-scale SCDPR in dynamic motion is an open challenge.**

This work proposes a real-time pose estimation algorithm for dynamic trajectories of SCDPRs, which is accurate over large areas. We present a novel approach that considers cable sag to reduce the estimation error for large scales while also employing an Inertial Measurement Unit (IMU) to improve estimation accuracy for dynamic motion. Our approach reduces the RMSE to less than a third compared to standard methods not considering cable sag. Similarly, the inclusion of the IMU reduces the RMSE in dynamic situations by 40% compared to non-IMU aided approaches considering cable sag. Furthermore, we evaluate our Extended Kalman Filter (EKF) based algorithm on a real system with ground truth pose information.

I. Introduction

Cable-Driven Parallel Robots (CDPR) have unique advantages over their rigid link counterparts. By replacing heavy rigid links with lightweight cables, the robot's inertia is drastically reduced, allowing for high acceleration dynamics and large workspaces since the maximum applicable tension is very high for such low weights of the cables. The costs and simplicity of the mechanical designs might be seen as key advantages for using cable robots as a replacement for conventional solutions in a growing number of industry applications. Application fields include production engineering [1], logistics [2], constructions [3], and motion simulation [4] [5].

When the workspace on the ground needs to stay clear of the cables, e.g., in production plants, the fully constrained CDPRs can not operate, and a viable alternative is the suspended configuration. For so-called Suspended Cable-Driven Parallel Robots (SCDPR), as shown in Fig. 1, all cables are attached above the moving platform, and gravity ensures that all cables are tensed, as shown in [2].

[1]Eren Allak, Christian Brommer and Stephan Weiss are with the Department of Smart Systems Technologies in the Control of Networked Systems Group, University of Klagenfurt, Austria {eren.allak, christian.brommer, stephan.weiss}@ieee.org

[2] Rooholla Khorrambakht is with Faculty of Electrical Engineering and Computer Science, K. N. Toosi University of Technology, Iran. r.khorrambakht@email.kntu.ac.ir

This research received funding from the Austrian Ministry of Climate Action and Energy (BMK) under the grant agreement 878661 (SCAMPI).

Pre-print version, accepted June/2022, 10.1109/IROS47612.2022.9981168 ©IEEE.

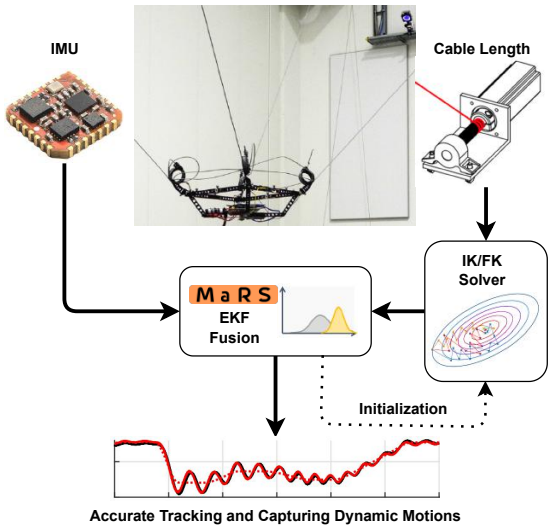

Fig. 1. The overall architecture of our proposed fusion framework. We use a suspended cable-driven robot with a relatively large workspace and a lightweight end-effector which leads to considerable cable sag. Our proposed kinematic solver together with an onboard IMU can accurately track the kinematically unobservable oscillations.

For an SCDPR to have three Degrees of Freedom (DoF) in translation, it must have at least three actuated cables. However, the operation of SCDPRs with four actuated cables enlarges the covered workspace and adds more safety to the system since an additional cable carries the platform while still being cost-efficient, as in this work.

The parallel actuation with cables comes at a price: Due to the existence of multiple kinematic loops in parallel robots, task-space pose estimation of such manipulators is more complex compared to their serial counterparts. These complications are even more prominent when these links are constructed using long cables that exhibit *cable sag*, i.e., under the influence of gravity and the negligible cable bending stiffness, the cables get a downwards curved shape.

The problem of estimating the position $\mathbf{p}_e$ and orientation $\mathbf{q}_e$ of an end-effector given the length of the cables is known as Forward Kinematics (FK). In contrast, the opposite problem of finding the cable length out of a given pose is called Inverse Kinematics (IK). In large scale suspended cable robots, the unavoidable cable sag necessitates a more complex kinetostatic analysis in solving the inverse and forward kinematics problems, and the system exhibits considerable dynamic oscillations before reaching equilibrium.

Accurate and low-latency tracking of the end-effector's 6-DoF motion is a requirement for effective control and numer-

ous applications such as robotic inspection [3] or augmented reality [4]. Even though applying onboard SLAM may be a solution to this problem, the unavoidable drift and sensitivity to visual depredations alongside the inherent high compute intensity prevents the easy adoption of such algorithms for proper loop-rate control and drift-free tracking.

In this paper, we combine the complementary advantages of a proposed equilibrium-state kinematic solver and a high-rate onboard MEMS Inertial Measurement Unit (IMU) using an Extended Kalman Filter (EKF). With this, we simultaneously track the transient dynamics using the IMU while providing a drift-free global pose constraint using the kinematics solver. Furthermore, we eliminate the necessity of loop-rate kinematics optimization execution by leveraging the high-rate IMU for motion propagation and optimizer initialization. To the best of our knowledge, this is the first real-time estimator to combine kinematics constraints considering cable sag with an IMU to achieve improved accuracy for large-scale cable robots. We provide our open-source implementations [1] and real dataset below.

Our contributions are:

- Presenting a configurable and efficient IK/FK solver for an underactuated large-scale suspended CDPRs with four sagging cables and based on a modern C++ optimization framework with automatic differentiation.
- Proposing a pose estimation framework based on an onboard IMU and our kinematic solver that can track the dynamic transients of the motion.
- Evaluation of the proposed method with real data on a field-proven commercial CDPR with ground truth pose information

II. Related Work

This section reviews the current state of research in cable robotics pose estimation. In general, the efforts for attaining accurate end-effector pose estimates may be broken down into the two categories of improving kinematic solvers and incorporating sensor fusion and auxiliary sensors. In what follows, we provide related works to these two directions and their relation to our proposed methodology.

A. Kinematic Solvers

Unlike medium-size cable robots, where the massless cable model assumption holds, the kinematic analysis of large-scale cable robots faces complexities. The effect of gravity on the cables leads to the sagging phenomenon, which complicates the kinematic analysis of the system. At the kinematics level, we are concerned with finding the pose of the robot in equilibrium state, and the works that address this problem can be broken down into two categories of real-time local solvers and slow but global algorithms that can find all possible solutions.

The full solution to the forward and inverse kinematics problems of cable robots with sagging cables has been considered by [6], [7], [8]. On the other hand, real-time solvers are often formulated based on the local refinement of the state using the latest solution as initialization [9] or based on special configurations that enable finding unique solutions in real-time [10]. Furthermore, real-time performance can also be achieved by restricting the solution space through the addition of a noiseless cable direction sensor as done in [11]. The idea of [11] is related to our formulation in the sense that we also greatly leveraged from splitting of cable directions to vertical and horizontal parts as discussed in III-A. However, we do not assume perfect cable directions. Instead, the end-effector poses and, therefore, the cable directions are iteratively optimized to achieve the best solutions.

Our forward kinematics solvers considering cable sag are similar to [9] [12]. However, we use a minimal singularity-free optimization variable for the end-effector pose, and our anchor points can be anywhere on the robot, and also, we propose a parametric solution for the force distribution in contrast to those works—all of the above in an ensemble lead to better convergence of the numerical solvers. Furthermore, we provide the open-source and highly efficient implementation of our solver based on the Ceres library [13] and with ROS and Python wrappers to the community together with the data from a real SCDPR.

B. Auxiliary Sensors and Sensor Fusion

The kinematics solvers introduced in the previous subsection do not model the transient and dynamical motions of the end-effector. Therefore, there is an upper limit on the achievable accuracy based on cable length measurements alone. Even though the dynamical modeling of the system for further improvements may also be considered, the highly complex computations required for doing so prevent real-time deployment. More importantly, the states of this dynamical model may not be observable from the encoder's point of view. Motivated by these limitations, a new emerging interest in the literature is to incorporate other sensors alongside the kinematic modalities to make up for the mutual limitations of using each sensor alone.

To the best of our knowledge, the authors of [14] were the first to use sensor fusion for the localization of cable robots. Their work is based on onboard vision and inertial sensors alongside the kinematics of the robot based on rigid cable assumption, which defines cables to be massless and inextensible. The vision relies on April-Tags with known locations and their IMU formulation is based on a dual-stage Kalman filter for estimating orientation and then propagating the positions.

More recently, [15] proposed a more elegant EKF-based IMU-Kinematics fusion framework that assumed the cables to be rigid, and showed improved tracking performance through Monte Carlo simulations. The authors of [16] and [17] present similar IMU-Kinematic fusion setups, respectively, for a suspended facade cleaner robot and a wire-based joystick system. Our work is most closely related to [15], but we go one step further. The novel accomplishment of our work is that we additionally consider the mathematically intricate sagging of the cables in the fusion with an IMU and

[1] https://github.com/aau-cns/scampi_ks_mars_fusion

validate our algorithm based on its real-time execution in a real-world setting.

III. Preliminaries and Definitions

In this section, we introduce the coordinate frames and how to address the 2D problem of finding catenaries in 3D with a recap on the theory and the boundary conditions we use.

A. Coordinate Frames and Cable Force Directions

We have three relevant coordinate frames: the global frame $\mathcal{G}$, the local end-effector frame $\mathcal{L}$ and the IMU frame $\mathcal{I}$. According to the following coordinate conventions $\mathbf{p}_e$ is defined as ${}_{\mathcal{G}}\mathbf{p}_{\mathcal{GL}}$. The left subscript is the frame in which the vector is expressed, and the right subscripts indicate that it points from the first frame to the second frame. In this case, the vector is expressed in $\mathcal{G}$ and points from the center of $\mathcal{G}$ to the center of $\mathcal{L}$. For the orientation $\mathbf{q}_e$ a Hamiltonian quaternion $\mathbf{q}_{\mathcal{GL}}$ is used, transforming vectors from $\mathcal{L}$ to $\mathcal{G}$. The IMU calibration pose $\{\mathbf{p}_{\text{imu}}, \mathbf{q}_{\text{imu}}\}$ is given as $\{{}_{\mathcal{I}}\mathbf{p}_{\mathcal{IL}}, \mathbf{q}_{\mathcal{IL}}\}$. For legibility we will omit the left subscript $\mathcal{G}$, such that vectors like $\mathbf{p}_a$ and $\mathbf{p}_b$ are in fact ${}_{\mathcal{G}}\mathbf{p}_a$ and ${}_{\mathcal{G}}\mathbf{p}_b$. While ${}_{\mathcal{L}}\mathbf{b}$ is the vector that goes from the center of the end-effector to the anchor, $\mathbf{p}_b$ is the anchor's global 3D position. The global position of the pulley is $\mathbf{p}_a$, which is a constant 3D point.

Considering that the catenary equation solves a 2D problem, 3D cable directions can be split into a horizontal and a vertical part. The direction of the cable on the global x-y-plane is defined to be the unit vector $\mathbf{e}_{\mathcal{C}} = \texttt{normalize}\left((\mathbf{p}_b)_{xy} - (\mathbf{p}_a)_{xy}\right)$ and it has the coordinate ${}_{\mathcal{C}}x$ that goes from the pulley $\mathbf{p}_a$ to the anchor $\mathbf{p}_b$ in positive direction. Cable forces are split, accordingly, in a horizontal part f_h along $\mathbf{e}_{\mathcal{C}}$, and a vertical part f_v in the z direction of the global frame. Since the cable forces are supposed to carry an end-effector, we will introduce a positive f_v as a lifting force along $+\mathbf{e}_z$ on the end-effector. The reaction force on the cable will then point downwards. And a positive f_h on the cable along $\mathbf{e}_{\mathcal{C}}$ will pull on the robot along $-\mathbf{e}_{\mathcal{C}}$.

B. Catenary Equations and Boundary Conditions

We first model the cable shape $z({}_{\mathcal{C}}x)$ under gravity and with inextensible cables in static equilibrium, see (1). Note, z is the cable height, and ${}_{\mathcal{C}}x$ is the cable coordinate as defined above. To find the overall cable shape, an ordinary differential equation on infinitesimally small cable segments and their force equilibrium has to be solved. The solution, as in [18], is shown:

$$z({}_{\mathcal{C}}x) = \frac{f_h}{g_c} \cdot \left(\cosh\left(\frac{g_c}{f_h} \cdot ({}_{\mathcal{C}}x + C_1)\right) - C_2\right) \quad (1)$$

$$z'({}_{\mathcal{C}}x) = \sinh\left(\frac{g_c}{f_h} \cdot ({}_{\mathcal{C}}x + C_1)\right) \quad (2)$$

The height $z({}_{\mathcal{C}}x)$ of the cable at ${}_{\mathcal{C}}x$ is aligned with the global z direction, while the direction of the cable was previously defined as $\mathbf{e}_{\mathcal{C}}$ (on the horizontal x-y-plane) with coordinate ${}_{\mathcal{C}}x$. Accounting for the weight force of the cable is done by $g_c = g \cdot \rho\prime_c$, where $\rho\prime_c$ is the weight per length of the cable and g is the gravitational constant. Since a differential equation was solved, C_1 and C_2 describe integration constants that can be found by using boundary conditions. The pulley $\mathbf{p}_a$ is located at ${}_{\mathcal{C}}x = 0$ and has the height $z(0) = (\mathbf{p}_a)_z$, while the anchor $\mathbf{p}_b$ is located at ${}_{\mathcal{C}}x = L = ||(\mathbf{p}_b)_{xy} - (\mathbf{p}_a)_{xy}||$ and has a height of $z(L) = (\mathbf{p}_b)_z$. Note that f_h is just the magnitude of the horizontal cable force, while the direction is defined by $\mathbf{e}_{\mathcal{C}}$. An important boundary condition at the end-effector is $z'(x) = \frac{-f_v}{f_h}$, ensuring that the total cable force has to have the same direction as the cable, remembering that the vertical force was introduced as a lifting force at the end-effector and therefore it has to point down at the cable. It can be seen that the cable shape $z({}_{\mathcal{C}}x)$ is coupled to the anchor position and the cable forces. That is why the kinematic and the static computations are coupled, and the presented solver will take this into account.

IV. The Structure of IK/FK Solvers and the Constituent Blocks

This section introduces the architecture of our kinematic solver and describes the computational building blocks that are used to define it. First, we present our optimization loop in IV-A. Then, cable force and catenary computation blocks are introduced in IV-C and IV-B, respectively. Finally, the optimization objectives for the forward and inverse Kinematics are presented in sections IV-E and IV-D, respectively.

A. The Structure of the Solver

We formulate our kinematic solver for underactuated suspended cable robots with four cables $\mathcal{C}_1, \mathcal{C}_2, \mathcal{C}_3$ and $\mathcal{C}_4$. The following parameters are considered to be known: pulley positions $\mathbf{p}_{a,i}$, relative anchor positions ${}_{\mathcal{L}}\mathbf{b}_i$, the mass of the end-effector m_e, and the cable weight parameter g_c. Furthermore, we know the relative center of gravity (COG) ${}_{\mathcal{L}}\mathbf{b}_{\text{cog}}$, pointing from the center of $\mathcal{L}$ to the COG. Finally, an initial position $\mathbf{p}_{e,\text{init}}$, orientation $\mathbf{q}_{e,\text{init}}$ and an initial force distribution $\{f_{h1,\text{init}}, f_{v1,\text{init}}\}$ are assumed to be given, as a starting point for the optimization. The values and the source of the initial estimates are discussed for the IK and FK in section V and the corresponding optimization objectives are defined in respective sections IV-D and IV-E.

Fig. 2 illustrates the architecture of our solver. All known/fixed parameters are fed to the blocks with blue arrows, and all resultant computations are outputted with black arrows. Starting from the initial values, in each iteration, the optimizer block (Optimization Step) modifies the optimization variables to yield a better set with lower cost and it feeds them back again into the blocks to iteratively improve the results, until convergence (i.e., no further decrease in cost function). We keep the pose variables as a 3-vector and a quaternion. Before optimizing, we transform the orientation quaternions $\mathbf{q}$ to $\mathbf{R}(\mathbf{q})$, which are elements of SO(3). During each optimization iteration, we find $\boldsymbol{\xi}_e$, a tangent space element of SO(3), to improve the orientation as $\mathbf{R}_{\text{new}} = \mathbf{R}_{\text{old}} \cdot \exp(\boldsymbol{\xi}_e)$. Using $\boldsymbol{\xi}_e$ and the exponential

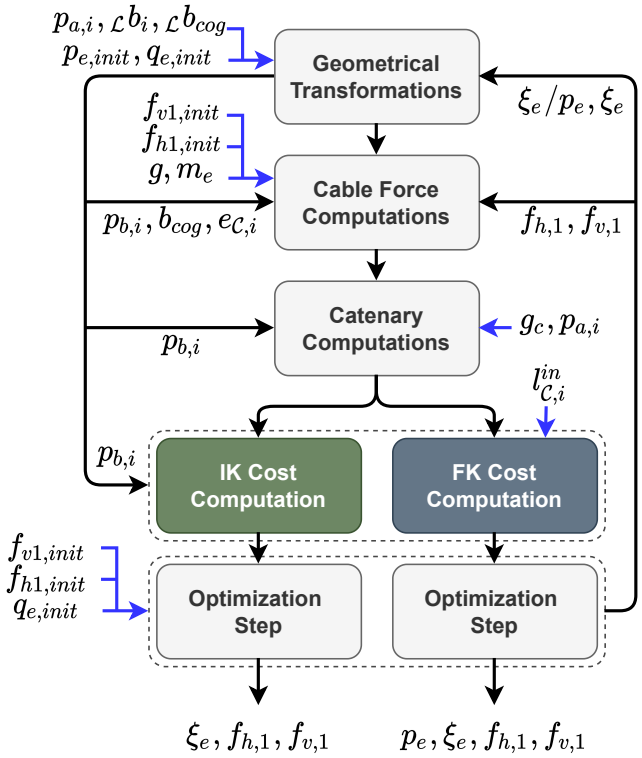

Fig. 2. The optimization loop of the proposed kinematic solver. Fixed parameters are fed to the blocks by blue arrows, and the resultant variables are exported with black arrows. For inverse or forward kinematics modes, only the corresponding left or right branches are activated.

map to get a correction for the orientation is a minimal and singularity-free representation (see [19]).

The first block in Fig. 2 (Geometrical Transformation) computes the parameters in the global frame and also provides cable force directions as discussed in III-B. The resulting variables of the first block are needed by all of the other blocks. The other main blocks of our kinematic solvers are presented in the following sections.

B. Force Computations Block

The outputs of this block are cable forces $\mathbf{f}_{\mathcal{C},i}$ that satisfy the static equilibrium conditions and are defined as a function of the first cable forces and the calibration parameters. Providing a parametric solution for the overall force distribution as a function of the first cable forces is a novel aspect for kinematics solvers.

1) The Structure matrix: The cable forces at the end-effector are given as $\mathbf{f}_{\mathcal{C},i} = -f_{h,i} \cdot \mathbf{e}_{\mathcal{C},i} + f_{v,i} \cdot \mathbf{e}_z$ where $\mathbf{e}_z$ is the unit vector in z direction. Also the x-y-plane directions of the cables are given as $\mathbf{e}_{\mathcal{C},i}$. Based on the force (3) and moment (4) equilibrium conditions, the structure matrix $\mathbf{A}^T$ may be defined which is the transposed Jacobian of the robot.

The wrench vector $\mathbf{w}_{ee}$ contains the gravitational force vector and the moment induced by the offset of the center of gravity with respect to the geometrical center. All vectors are expressed in $\mathcal{G}$, therefore it should be noted that ${}_{\mathcal{L}}\mathbf{b}_i$ and ${}_{\mathcal{L}}\mathbf{b}_{\text{cog}}$ are transformed into the $\mathcal{G}$ frame first.

$$\mathbf{0} = \sum_{i=1}^{4} \mathbf{f}_{\mathcal{C},i} - m_e \cdot g \cdot \mathbf{e}_z \tag{3}$$

$$\mathbf{0} = \sum_{i=1}^{4} \mathbf{b}_i \times \mathbf{f}_{\mathcal{C},i} - \mathbf{b}_{\text{cog}} \times m_e \cdot g \cdot \mathbf{e}_z \tag{4}$$

$$\mathbf{w}_{ee} = \mathbf{A}^T \cdot \begin{bmatrix} f_{h,1} & f_{v,1} & f_{h,2} & f_{v,2} & \dots \end{bmatrix}^T \tag{5}$$

$$\mathbf{w}_{ee} = \mathbf{A}^T \cdot \mathbf{f}_{\mathcal{C}} = \begin{bmatrix} m_e \cdot g \cdot \mathbf{e}_z \\ \mathbf{b}_{\text{cog}} \times m_e \cdot g \cdot \mathbf{e}_z \end{bmatrix} \tag{6}$$

$$\mathbf{A}^T_{1,2} = \begin{bmatrix} -\mathbf{e}_{\mathcal{C},1} & \mathbf{e}_z & -\mathbf{e}_{\mathcal{C},2} & \mathbf{e}_z \\ -\mathbf{b}_1 \times \mathbf{e}_{\mathcal{C},1} & \mathbf{b}_1 \times \mathbf{e}_z & -\mathbf{b}_2 \times \mathbf{e}_{\mathcal{C},2} & \mathbf{b}_2 \times \mathbf{e}_z \end{bmatrix} \tag{7}$$

$$\mathbf{A}^T_{3,4} = \begin{bmatrix} -\mathbf{e}_{\mathcal{C},3} & \mathbf{e}_z & -\mathbf{e}_{\mathcal{C},4} & \mathbf{e}_z \\ -\mathbf{b}_3 \times \mathbf{e}_{\mathcal{C},3} & \mathbf{b}_3 \times \mathbf{e}_z & -\mathbf{b}_4 \times \mathbf{e}_{\mathcal{C},4} & \mathbf{b}_4 \times \mathbf{e}_z \end{bmatrix} \tag{8}$$

$$\mathbf{A}^T = \begin{bmatrix} \mathbf{A}^T_{1,2} & \mathbf{A}^T_{3,4} \end{bmatrix} \tag{9}$$

2) Parametric Solution to Force Distribution: We have eight unknown cable forces and only six equations from the force and moment equilibrium constraints. To address this problem, we parametrically define six of the eight cable force variables based on the other remaining two. With this, we effectively reduce the number of unknown forces from eight to two. An advantage of this novel approach is that the parametric solution will always satisfy the static equilibrium conditions, and the solver does not need to consider static equilibrium in its cost function.

$$\mathbf{w}_{ee} = \begin{bmatrix} \mathbf{A}^T_1 & \mathbf{A}^T_{2:4} \end{bmatrix} \cdot \begin{bmatrix} \mathbf{f}_{\mathcal{C},1} \\ \mathbf{f}_{\mathcal{C},2:4} \end{bmatrix} \tag{10}$$

$$= \mathbf{A}^T_1 \cdot \mathbf{f}_{\mathcal{C},1} + \mathbf{A}^T_{2:4} \cdot \mathbf{f}_{\mathcal{C},2:4} \tag{11}$$

$$\mathbf{f}_{\mathcal{C},2:4} = (\mathbf{A}^T_{2:4})^{-1} \left(\mathbf{w}_{ee} - \mathbf{A}^T_1 \cdot \mathbf{f}_{\mathcal{C},1} \right) \tag{12}$$

In Eq. (10) we split the structure matrix $\mathbf{A}^T$ in two parts. $\mathbf{A}^T_1$ is a 6-by-2 matrix of the first two columns and the remaining six columns form a 6-by-6 matrix $\mathbf{A}^T_{2:4}$. $\mathbf{f}_{\mathcal{C},1}$ is build from the first two entries of $\mathbf{f}_{\mathcal{C}}$ being $f_{h,1}$ and $f_{v,1}$. Now all other cable forces are described by the horizontal and vertical force components of the first cable as described in Eq. (12).

C. Catenary Computations Block

To compute the catenary variables we need the force distribution and the global poses of the anchors and pulleys. The catenary constants C_1 and C_2 for each cable $\mathcal{C}_i$ (note, calligraphic C) are found by applying the boundary conditions $z'(L) = \frac{-f_v}{f_h}$, i.e. total force direction along cable direction, and height of the pulley $z(0) = (\mathbf{p}_a)_z$, respectively. The length on the x-y-plane is again found as $L = ||(\mathbf{p}_b)_{xy} - (\mathbf{p}_a)_{xy}||$. The cable lengths $l_{\mathcal{C},i}$ as needed by the cost function blocks and are also derived here.

$$C_{1,i} = \frac{f_{h,i}}{g_c} \cdot \text{asinh}\left(\frac{-f_{v,i}}{f_{h,i}} \right) - L_i \tag{13}$$

$$C_{2,i} = \cosh\left(C_{1,i} \cdot \frac{g_c}{f_{h,i}} \right) - \frac{g_c}{f_{h,i}} \cdot (\mathbf{p}_a)_z \tag{14}$$

$$l_{\mathcal{C},i} = \int_0^{L_i} \sqrt{1 + \left(\frac{dz}{dx}\right)^2}\, dx \tag{15}$$

$$= \frac{f_{h,i}}{g_c} \cdot \left(\sinh\left(\frac{g_c}{f_{h,i}} \cdot (L_i + C_{1,i})\right) - \sinh\left(\frac{g_c}{f_{h,i}} \cdot C_{1,i}\right)\right) \tag{16}$$

D. Inverse Kinematics Cost Computations Block

For the case of inverses kinematics, the inverse cost is activated in the optimization loop shown in figure Fig. 2. Given a position $\mathbf{p}_e$, approximate forces of the first cable $\{f_{h1,\text{init}}, f_{v1,\text{init}}\}$ and an approximate orientation $\mathbf{q}_{e,\text{init}}$, IK aims to find the cable length, the force distribution and the catenary variables. While the position can be fixed, the orientation needs to be optimized. We chose to leave the orientation as an optimization variable due to the fact that in an underactuated configuration like ours, not every combination of position and orientation is feasible.

The cost term $\tilde{z}(L_i)$ penalizes the difference between the height $(\mathbf{p}_{b,i})_z$ of the anchor i computed from the end-effector pose and calibration parameters and the expected height of the catenary at that point $z(L_i)$ and starting from pulley i. The smaller $\tilde{z}(L_i)$ is, the more consistent the end-effector pose estimate is with the cables attached to the mobile platform. The cost term $\tilde{l}_i$ penalizes the difference between the sagging cable length and the straight line distance to emphasize the maximal possible tension of all cables, but still being in static equilibrium. The shorter the length of the sagging cable the higher the tension on it and the closer it is to the straight line. The last term $\exp(-f_{h,i})$ forces the unconstrained solver to only consider positive forces in the horizontal direction. In theory, this term could bias the solver to find high magnitudes for the horizontal forces. But, the cost reduction decreases exponentially fast towards zero such that we do not see an effect in practice. We only observed that the solver can try to find negative horizontal forces in the case of IK while this is not the case for FK. As weights we report $\lambda_1 = 500$ and $\lambda_2 = 10$.

$$\tilde{z}(L_i) = (\mathbf{p}_{b,i})_z - \frac{f_{h,i}}{g_c} \cdot \left(\cosh\left(\frac{g_c}{f_{h,i}} \cdot (L_i + C_{1,i})\right) - C_{2,i}\right) \tag{17}$$

$$\tilde{l}_i = l_{\mathcal{C},i} - ||\mathbf{p}_{a,i} - \mathbf{p}_{b,i}|| \tag{18}$$

$$K_{\text{IK}} = \lambda_1 \cdot \sum_{i=1}^{4} \tilde{z}(L_i)^2 + \lambda_2 \cdot \sum_{i=1}^{4} \tilde{l}_i^2 + \sum_{i=1}^{4} \exp(-f_{h,i}) \tag{19}$$

E. Forward Kinematics Cost Computations Block

For the case of forward kinematics, only the forward cost is activated in the optimization loop shown in Fig. 2. Given an initial approximate pose $\{\mathbf{p}_{e,\text{init}}, \mathbf{q}_{e,\text{init}}\}$ and approximate forces of the first cable $\{f_{h1,\text{init}}, f_{v1,\text{init}}\}$, FK optimizes for both the orientation $\mathbf{q}_e$ *and* the position of the end-effector $\mathbf{p}_e$ given the measured cable lengths $l^{\text{in}}_{\mathcal{C},i}$.

The cost function is constructed similar to the IK problem. In fact, the first term $\tilde{z}(L_i)$ is defined as in Eq. (17). On the other hand, $\tilde{l}_i$ is defined as the difference between the cable lengths from the solver ,$l_{\mathcal{C},i}$, and the measured $l^{\text{in}}_{\mathcal{C},i}$ counterparts. The smaller $\tilde{l}_i$ is, the more consistent are the cable length measurements with the expected cable lengths computed from the optimized variables. As weights we report $\lambda_3 = 10$ and $\lambda_4 = 10$.

$$\tilde{l}_i = l_{\mathcal{C},i} - l^{\text{in}}_{\mathcal{C},i} \tag{20}$$

$$K_{\text{FK}} = \lambda_3 \cdot \sum_{i=1}^{4} \tilde{z}(L_i)^2 + \lambda_4 \cdot \sum_{i=1}^{4} \tilde{l}_i^2 \tag{21}$$

V. Kinematics-Inertial Fusion

The previous section presented our IK and FK solvers. In this section, we use *both* solvers together to compute an end-effector pose from cable lengths and fuse it with the IMU to provide real-time pose estimates. As shown in 1, our solvers collectively function as a *pose sensor* providing updates in a multi-sensor fusion framework [20] based on Extended Kalman Filtering (EKF), while the IMU is used as a *propagation sensor*. In the beginning, the solvers are initialized once with a good initial guess (in our experiments with a ground truth pose). After this, the solver's output is fed to the estimator framework as a pose update. The measurement model for pose updates with error quaternions, the correction step, and the Jacobians for the EKF are outlined in [21]. With this pose update, the estimator corrects the previous pose estimate and propagates the pose with new IMU readings, i.e., integrates the acceleration and the rotational velocity to get new poses, as described in the following. These propagated poses enable us to track the dynamic motion of the end-effector, but pure IMU integration quickly diverges due to miss-alignment towards gravity and sensor noise. Thus the integration needs to be corrected frequently. Once new cable length measurements are available to the solver, the latest propagated pose of the estimator is used as an initial pose for the solver (shown with a dashed arrow in Fig. 1). The solvers then compute a pose given the cable lengths and the whole update and propagation cycle repeats.

The measurement noise must account for errors between the measured pose, i.e. static equilibrium pose, and the pose during oscillations. Although this error distribution is non-gaussian, we achieved accurate tracking using a fixed noise covariance.We chose the values big enough to allow the tracking with the propagated estimate but also small enough to prevent drifting estimates.

In theory, we would only need to use the FK solver, but it requires a good initial guess of the force distribution, which we do not measure. On the other hand, our IK solver robustly finds good force distributions given a naive guess of the forces of the first cable and a 3D position. We assume for our naive guess $f_{v,1} = 1/4 m_p g$ and $f_{h,1} = f_{v,1}/\tan(\alpha)$, where α is the angle between the straight line connecting anchor and pulley and the ground plane.

Another aspect to consider is the numerical stability of the solvers. Since we have the first cable forces $f_{h,1}$ and $f_{v,1}$ as optimization variables, the choice which cable is the *first* cable has an impact. Choosing the cable over the longest

distance as the *first* leads to improved numerical stability of the solvers.

We are working with a recursive filtering framework named MaRS [20] that allows run-time self-calibration and is computationally efficient. Assuming an IMU for propagation, the core-states are: Position expressed in the global frame $\mathbf{p}_{\mathcal{GI}}$, velocity w.r.t. the global frame $\mathbf{v}_{\mathcal{GI}}$, orientation of the robot w.r.t. the global frame $\mathbf{q}_{\mathcal{GI}}$ as well as IMU biases for the gyroscope $\mathbf{b}_\omega$ and accelerometer $\mathbf{b}_a$. The core state and dynamics as in [21] are described as follows:

$$\dot{\mathbf{p}}_{\mathcal{GI}} = \mathbf{v}_{\mathcal{GI}} \tag{22}$$

$$\dot{\mathbf{v}}_{\mathcal{GI}} = \mathbf{R}_{(\mathbf{q}_{\mathcal{GI}})}(\mathbf{a}_m - \mathbf{b}_a - \mathbf{n}_a) - \mathbf{g} \tag{23}$$

$$\dot{\mathbf{q}}_{\mathcal{GI}} = \frac{1}{2}\mathbf{\Omega}(\boldsymbol{\omega}_m - \mathbf{b}_w - \mathbf{n}_w)\mathbf{q}_{\mathcal{GI}} \tag{24}$$

$$\dot{\mathbf{b}}_\omega = \mathbf{n}_{\mathbf{b}_\omega},\ \dot{\mathbf{b}}_a = \mathbf{n}_{\mathbf{b}_a} \tag{25}$$

The IMU can also be disabled and the FK solver can run in standalone mode. This mode uses the last FK pose to initialize the current optimization to find a pose for new cable length measurements. With our current configuration, as described in the next section, we can run our FK solver at a rate of up to 300Hz, but based on our experiments and for our data with moderate end-effector speeds, even when running at 10 Hz, it still provides accurate pose estimates as shown in VI.

VI. Experiments

A. Setup

We work with a commercial cable robot actuated by four cables in a suspended configuration with a noticeable cable sag. The cables go out of winches fixed on the floor upwards to pulleys attached to the walls, that are considered as the static anchor locations. The cables then connect the pulleys to the rigid anchor points on the end-effector. The rectangular configuration of pulleys attached at the height of 8.5 m span a field of 4.5 m in x and 12.5 m in y direction. Since the y dimension of the field is almost three times larger than the x direction, the robot is very stiff in y direction and as a result, we encounter more significant oscillations in the x direction during the operation, that necessitate proper tracking of the pose and accounting for dynamic motions.

The task-space of the robot is constrained to a smaller region of $1\text{m} \times 4\text{m} \times 4\text{m}$ for safety reasons. As shown in Fig. 1, the end-effector is structured as a truncated pyramid. The IMU is attached on the smaller bottom and the anchors are located on the bigger top side with edge lengths of 0.21 m. The whole end-effector weighs 4.4 kg in total, and the mass per length for the used aramid cables is 10.55 g/m. We use an X-Sense MTi-100 MEMS[2] IMU running at 400 Hz and the winch encoders used for relative length measurements provide readings at a rate of 125 Hz. A motion tracking system that we use for ground truth poses measures the movement of the robot platform and the pulleys with tracked markers up to 1 mm accuracy.

[2]Micro Electro Mechanical Systems

We have implemented our solver based on the Google Ceres optimization library [13]. We exploit the automatic differentiation capability of this library to avoid the computation of analytical Jacobian matrices without sacrificing the performance and numerical stability associated with numerical differentiation. Based on this implementation, the back-to-back execution of our solvers take on average $2.3ms$ per inference on a computer equipped with a Core-i7 7700HQ CPU and 32GB or RAM. Furthermore, we have implemented our fusion framework by wrapping our solver as a ROS node and have used the MARS EKF framework for the fusion engine. For the benefit of the community, our implementations will be made available as open-source packages.

The procedure of our experiments is to move to 40 random points, with random waiting times of a few seconds on each point. During sudden acceleration and decelerations, the robot starts to oscillate around its equilibrium point. As we show in this section, we can track these dynamic oscillations by fusing the IMU with our FK solutions.

B. Pose Estimation with Oscillations

We conducted a series of experiments with random point trajectories, where at each point, the robot stops for two to three seconds. These sudden velocity changes after traveling at a speed of 0.6 to 1.0 m/s lead to oscillations of the end-effector. We observed maximal oscillation amplitudes of 20 cm, 5 cm, and less than 1 cm in the x, y, and z direction, respectively. The maximally observed amplitudes in the orientation are $2.5°$, $10°$ and $2°$ in roll, pitch, and yaw, respectively. The reason for the strong oscillations in the x direction and pitch (rotation around x) can be explained by the stretched pulley configuration exerting strong stabilizing forces along y, but not x. In Fig. 3 and Fig. 4 we show that the pose is tracked even out-of-equilibrium conditions with the fusion of the IMU for a sample trajectory. Although we can track transient motion with IMU propagation, pose estimation purely based on IMU is subject to quick accumulation of errors and a constraining modality is essential to prevent that. For this purpose, we employ our FK solver and estimate the pose achieving accurate estimates during motion and equilibrium.

The estimation error, as seen in Fig. 3 and Fig. 4 bottom plots, seems to scale with the amount of motion, especially during strong oscillations. As expected, stronger oscillations are farther away from equilibrium conditions and rely more on the noisy IMU. Nevertheless, we also observe that the estimate gets more accurate as oscillations fade out. Regarding the orientation, we can see in the sample trajectory that the initial wrong yaw is corrected as soon as the robot starts moving since the FK orientation and the global position render the heading of the robot observable. Our global position updates are an advantage that we have over SLAM. As seen in the plots, our yaw and position errors remain bounded throughout the trajectory, as a result of this globally consistent modality.

In Fig. 5 we analyze two cases of motion in more detail: the strongest and the weakest oscillations of the sample trajectory are shown in 3, in the top two plots and the bottom two plots, respectively. Only the x direction is shown for clarity since the oscillations are about ten times stronger in x than in y and z. While our estimator, i.e., the fusion of IMU and FK solver, can accurately track the dynamic trajectory, the output of the FK solver in standalone mode follows the center of the oscillations since the equilibrium point is expected to be there.

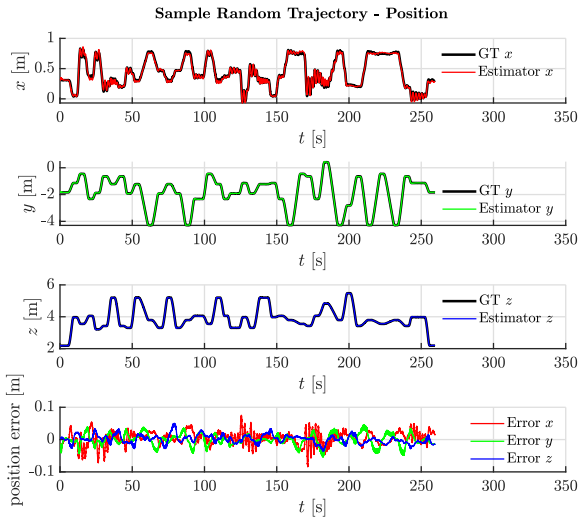

Fig. 3. We show the estimated positions compared to the ground truth (black) and the corresponding errors. While the strongest oscillations have a magnitude of 20 cm, our RMSE error is around 3 cm.

C. Forward Kinematics with and without Cable Sag

Our FK solver can find equilibrium points throughout the trajectory when cable measurements are given. The output of our solver in standalone mode and the ground truth is shown in Fig. 6 alongside commercial software that assumes straight lines. Since the robot is swinging most of the time, the exact equilibrium point is unknown. However, assuming that the equilibrium lies precisely between the amplitude peaks, we demonstrate that our solver finds the equilibrium in Fig. 6. The commercial software is off by a considerable margin with roughly 10 cm at max in x and 2-5 cm in y. The errors in z are about 1 cm for both. The zoomed plot in Fig. 6 visualizes the errors in y.

Another essential feature of the proposed FK solver is that based on our practical observations, even feeding initial poses far from the equilibrium as initial values to the solver, we still converge to the same equilibrium points. There is transient motion most of the time, and therefore, the estimator pose is far from the equilibrium state. Feeding the estimator pose during oscillations in the solver and still getting the same equilibrium poses as output shows the robustness of our solver and that it is applicable during dynamic motion.

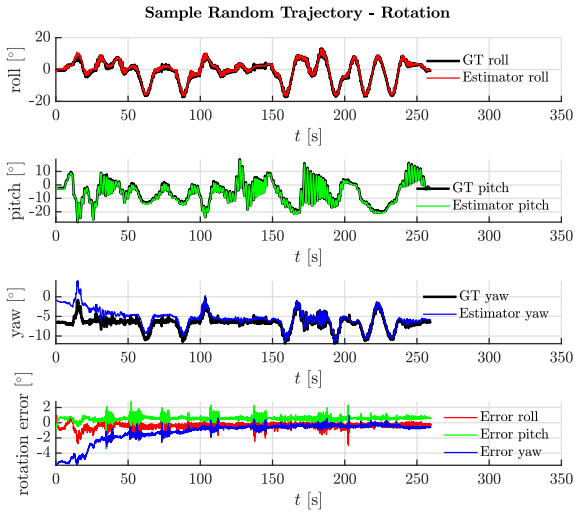

Fig. 4. We show the estimated orientation compared to the ground truth (black) and the corresponding errors. Since our solver also provides a rotation estimate the estimated orientation does not drift, as it would be the case for IMU fused with only global position.

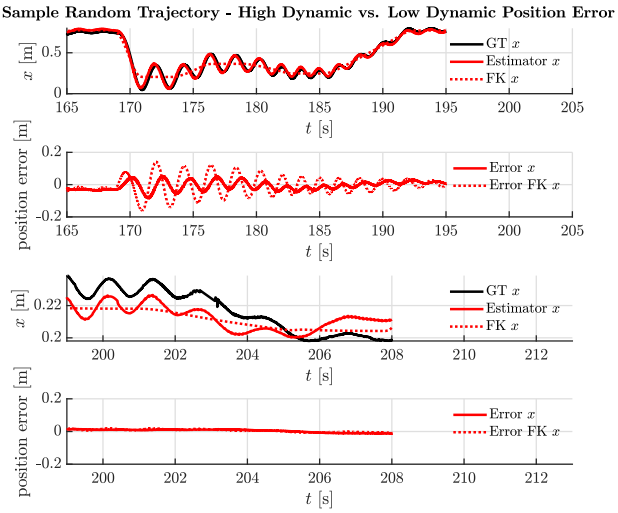

Fig. 5. A section of the trajectory where the end-effector undergoes dynamic osculations that are unobservable from solver alone. The fusion of IMU data leads to the reduction of errors and observing those oscillations.

TABLE I

THE QUANTITATIVE RESULTS OF OUR FRAMEWORK COMPARED TO VARIOUS BASELINES

Experiment	RMSE [m]
SL Solver	0.0693
FK Solver	0.0436
FK Solver + IMU	**0.0313**
FK Solver (high dynamic)	0.0610
FK Solver + IMU (high dynamic)	**0.0369**
FK Solver (low dynamic)	0.0270
FK Solver + IMU (low dynamic)	**0.0217**

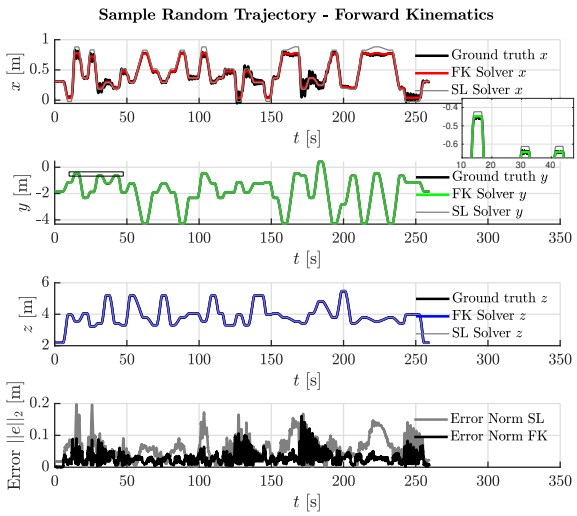

Fig. 6. Estimated poses by the solver compared to the ground-truth and the output of a naive forward kinematics algorithm based on straight line cables attached to a single point on the end-effector (SL Solver). As expected, considering cable sag in the solver (FK solver) improves the pose estimates.

VII. Conclusions

We presented a modular solver to compute the equilibrium-state pose of the end-effector in underactuated suspended cable robots with four sagging cables. We formulate our solver using a minimal and singularity-free representation of the pose and present a novel reparametrization approach for the cable forces that automatically ensure the static equilibrium constraints during the optimization.

Implemented in C++ and with automatic differentiation, our efficient solver works alongside a high-rate IMU within an EKF-based fusion backend to allow the tracking of out-of-equilibrium dynamic motions. This bi-lateral approach of considering cable sag and fusing with an IMU is the novel aspect of our work, and our practical evaluations based on a real-world system and accurate ground truth pose measurements verify the applicability of our method in real-world scenarios. We demonstrated that our algorithm can successfully track the swinging motions of the cable robot, which otherwise would have been unobservable based on a forward kinematics modality alone. This opens the possibility for a range of new applications since accurate localization, even in dynamic conditions and on a large-scale robot, is an elemental step for autonomous operation of SCDPRs.

Acknowledgement

The authors would like to thank Daniel Frumlacher for the committed and solution-driven approach during the data acquisition and for the handy implementations of a variety of sensors on the cable robot.

References

[1] "Parallel cable robotics for improving maintenance and logistics of large-scale products." [Online]. Available: www.cablebot.eu

[2] J. Lamaury and M. Gouttefarde, "Control of a large redundantly actuated cable-suspended parallel robot," in *2013 IEEE International Conference on Robotics and Automation*, 2013, pp. 4659–4664.

[3] A. Pott, C. Meyer, and A. Verl, "Large-scale assembly of solar power plants with parallel cable robots," in *ISR 2010 (41st International Symposium on Robotics) and ROBOTIK 2010 (6th German Conference on Robotics)*, 2010, pp. 1–6.

[4] P. Miermeister, M. Lächele, R. Boss, C. Masone, C. Schenk, J. Tesch, M. Kerger, H. Teufel, A. Pott, and H. H. Bülthoff, "The CableRobot simulator large scale motion platform based on cable robot technology," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2016, pp. 3024–3029.

[5] J. v. Zitzewitz, G. Rauter, R. Steiner, A. Brunschweiler, and R. Riener, "A versatile wire robot concept as a haptic interface for sport simulation," in *2009 IEEE International Conference on Robotics and Automation*, 2009, pp. 313–318.

[6] J.-P. Merlet, "A generic numerical continuation scheme for solving the direct kinematics of cable-driven parallel robot with deformable cables," in *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2016, pp. 4337–4343.

[7] ——, "On the inverse kinematics of cable-driven parallel robots with up to 6 sagging cables," in *2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2015, pp. 4356–4361.

[8] ——, "A new generic approach for the inverse kinematics of cable-driven parallel robot with 6 deformable cables," in *Advances in Robot Kinematics 2016*, J. Lenarčič and J.-P. Merlet, Eds. Cham: Springer, 2018, pp. 209–216.

[9] J.-P. Merlet and J. Alexandre-dit Sandretto, "The forward kinematics of cable-driven parallel robots with sagging cables," in *Cable-Driven Parallel Robots*, A. Pott and T. Bruckmann, Eds. Cham: Springer, 2015, pp. 3–15.

[10] J.-P. Merlet, "The forward kinematics of the 4-1 cable-driven parallel robot with non elastic sagging cables," in *Advances in Robot Kinematics 2020*, J. Lenarčič and B. Siciliano, Eds. Cham: Springer, 2021, pp. 98–108.

[11] ——, "Direct kinematics of CDPR with extra cable orientation sensors: The 2 and 3 cables case with perfect measurement and sagging cables," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2017, pp. 6973–6978.

[12] N. Riehl, M. Gouttefarde, S. Krut, C. Baradat, and F. Pierrot, "Effects of non-negligible cable mass on the static behavior of large workspace cable-driven parallel mechanisms," in *2009 IEEE International Conference on Robotics and Automation*, 2009, pp. 2193–2198.

[13] S. Agarwal, K. Mierle, and Others, "Ceres solver." [Online]. Available: http://ceres-solver.org

[14] D. Theodorakatos, E. Stump, and V. Kumar, "Kinematics and pose estimation for cable actuated parallel manipulators," in *International Design Engineering Technical Conferences and Computers and Information in Engineering Conference*, 2007, pp. 1053–1062.

[15] V. Le Nguyen and R. J. Caverly, "Cable-driven parallel robot pose estimation using extended kalman filtering with inertial payload measurements," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 3615–3622, 2021.

[16] M. Choi, M. Seo, H. S. Kim, and T. Seo, "Ukf-based sensor fusion method for position estimation of a 2-dof rope driven robot," *IEEE Access*, vol. 9, pp. 12 301–12 308, 2021.

[17] E. Yavuz, Y. Şenol, M. Özçelik, and H. Aydın, "Design of a string encoder-and-IMU-based 6D pose measurement system for a teaching tool and its application in teleoperation of a robot manipulator," *Journal of Sensors*, 2021.

[18] A. Pott, *Cable-Driven Parallel Robots*. Cham: Springer, 2018.

[19] J. Solà, J. Deray, and D. Atchuthan, "A micro lie theory for state estimation in robotics," 2018. [Online]. Available: https://arxiv.org/abs/1812.01537

[20] C. Brommer, R. Jung, J. Steinbrener, and S. Weiss, "MaRS: A modular and robust sensor-fusion framework," *IEEE Robotics and Automation Letters*, vol. 6, no. 2, pp. 359–366, 2021.

[21] S. Weiss and R. Siegwart, "Real-time metric state estimation for modular vision-inertial systems," in *2011 IEEE International Conference on Robotics and Automation*, 2011, pp. 4531–4537.