2024 IEEE International Conference on Robotics and Automation (ICRA 2024)
May 13-17, 2024. Yokohama, Japan

Adaptive State Estimation with Constant-Curvature Dynamics Using Force-Torque Sensors with Application to a Soft Pneumatic Actuator

Maximilian Mehl[1], Max Bartholdt[1,2,3], Simon F. G. Ehlers[3], Thomas Seel[3] and Moritz Schappler[3]

***Abstract*— Using compliant materials leads to continuum robots undergoing large deformations. Their nonlinear behavior motivates the use of model-based controllers. They require state estimation as an essential step to be deployed. Available sensors are usually realized by introducing rigid bodies to the soft robot or inserting soft sensors made of materials different from the robot itself. Both approaches result in changes in the system's dynamics. Optical measurements are problematic, especially in confined spaces. This can be avoided when the sensor is located at the robot's base. This paper studies the state estimation of a pneumatically actuated soft robot using the measured forces and torques at its base. For the first time, this is done using an unscented Kalman filter without restraining the dynamics to a planar or quasi-static motion while applying it to a real system. Real-time capability is achieved with our implementation. The state estimation is tested in a Cosserat rod simulation and on the physical system. The position is estimated with an accuracy of three to five millimeters for a 130 millimeter long pneumatic robot.**

I. Introduction

Using soft pneumatic actuators (SPA) poses the challenging task of accurate trajectory tracking. Sole model-based feedforward action suffers from inaccuracies due to the system's complexity. A common strategy to overcome the challenge is to utilize feedback control strategies known from rigid robots [1]. As these controllers rely on suitable feedback information on the system's current states, sensors matching to the task have to be deployed. In constrained settings, partial occlusion hinders the effective operation of vision-based tracking systems. Additionally, rigid optical markers have to be attached to the robot. Other sensors suffer from introducing rigid bodies or additional materials into the soft system. This includes inertial measurement units (IMU) [2], cameras attached to the robot [3], optical waveguides [4], [5], and flex sensors [6]. In [7] it is mentioned that sensors should alter the system's dynamics as little as possible. These problems can be avoided by utilizing measurements outside the deforming system, at the robot's base. Previous applications have incorporated base measurements for shape estimation of continuously deforming systems in the (quasi-) static case [8], [9], [10], [11], or for parameter identification [12], [13]. While satisfying results were obtained for shape estimation in quasi-static experiments, dynamic movements were omitted. Thus, the gap of *state estimation by evaluating the base reactions* concerning rapid motions remains.

Funded by the Deutsche Forschungsgemeinschaft (DFG, German Research Foundation) under grant no. 405032969.

[1] Both authors contributed equally to this work.

[2] Corresponding author.

[3] The authors are with the Institute of Mechatronic Systems, Leibniz University Hannover

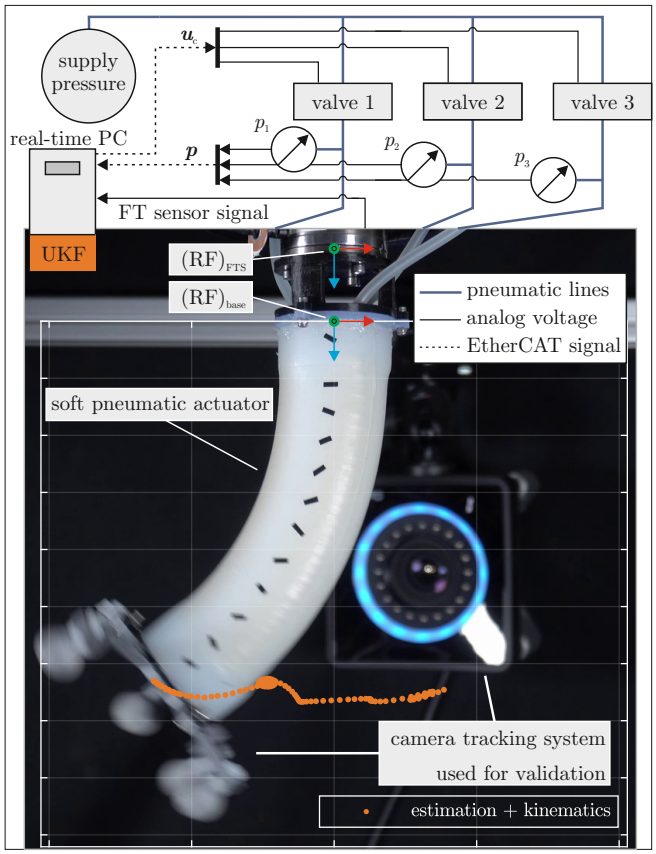

Fig. 1. Overview of the test bench and the proposed method, with a qualitative preview of the performed estimation. Three pneumatic air chambers are actuated as displayed. The pressure p_i of each chamber is measured relative to environmental pressure at the pneumatic supply line. Together with the force-torque (FT) signal, an estimation of the robot's configuration (orange) is performed using an unscented Kalman filter (UKF).

A. Related Work on state estimation in soft robots

We propose using an unscented Kalman filter (UKF) to reach this goal. Kalman filtering has found several use-cases in the soft robotics context. Sensor fusion for a human finger is done via Kalman filter using vision-based tracking combined with a flex sensor in [14] and by an extended Kalman filter (EKF) combining IMUs and draw-wire sensors in [15]. An EKF is used in [16] using optical fibers as input to measure the steady state. In [17], a dynamics and kinematics model is used with an EKF in a simulation environment. The application of a UKF is demonstrated in [18], using flex sensors and a neural network. In [19], state observation is done by a UKF using flex and pressure sensors with a second-order transfer function to model the system. A

dual EKF (parameter and state estimation) is investigated in simulation studies [20] using position and orientation measurements along the robot for the 2D case. In [21], an EKF is also used for estimating the wrench at the robot's tip. The results obtained show the applicability of Kalman filters in soft robotics, but the easily measurable base reaction forces and torques have not yet been incorporated as measurement feedback. Other estimation techniques incorporate end-effector pose and internal strains in statics [22] or sensed tip velocities [23]. However, the base reaction forces and torques are not incorporated in these works as well.

B. Contribution

For the first time, the robot's state in the dynamic case, consisting of deformation and its rate of change, is estimated by evaluating only the forces and torques at the system's base and the actuation input. The presented technique estimates an SPA's motion, consisting of both bending and elongation. To fully capture the system's behavior, without confining it to quasi-static or 2D scenarios, the analysis encompasses dynamic 3D motion. Real-time estimation is possible through the usage of a computationally cheap constant-curvature (CC) model. This allows for solving the state-estimation problem by incorporating a UKF, which is widely used for nonlinear systems. By including the bending stiffness in the state vector, the error arising from the kinematic assumption of CC is mitigated. The proposed filter is evaluated on data from a Cosserat rod simulation and on data experimentally obtained from the real system.

The contributions of this paper are:

- A model of SPA dynamics using CC kinematics suited for Kalman filtering using the base-reaction forces and torques,
- demonstrating the applicability of the proposed UKF on simulation and experimental data,
- evaluating real-time properties of the proposed filter.

The remainder of the paper is structured as follows: First, the modeling of the system's dynamics together with the filter is described in Sec. II. A demonstration of the UKF on a real SPA (Sec. III) and simulation (Sec. IV) follows with a brief discussion. The paper concludes with a summary of the results and an outlook to possible future work on the Kalman filtering using base-reaction forces and torques in Sec. V.

II. MODELING

The modeling of the kinematics and dynamics of the SPA is briefly described in Sec. II-A. The permutation of the model into a state-space representation follows in Sec. II-B. Finally, a discretized model tailored for Kalman filtering is presented in Sec. II-C.

A. Kinematics and Dynamics

To construct the model, a floating-base formulation [13], [24], a state-of-the-art formulation of CC kinematics [25], and a lumped mass in the modeled center of gravity [26] are combined. The modeling follows our previous work in [13], and is briefly summarized here.

As real-time ability for online state estimation is required, the deformation of the SPA is modeled as circular arc. This leads to the well-known formulation of CC kinematics. The kinematic state of the deforming system thus can be described by using only three CC parameters. A shortcoming of classical formulations of CC kinematics is a singularity of the formulation in the straight configuration. This is avoided by using the parametrization presented in [25]. The arc is described by $\boldsymbol{q}_\mathrm{r} = [\Delta_x, \Delta_y, \delta L]^\mathrm{T}$, with δL describing the elongation w.r.t. the reference length L_0, and Δ_x and Δ_y describing the bending angle and plane by defining length differences resulting from the pose of the tip w.r.t. the base. Additionally, the bending parameters include a free scaling parameter, here chosen as the SPA's outer radius r. The kinematic assumption for the deformation is combined with a floating (freely moving) base. Thereby, the forces and torques at the system's base are included in the model. Because the real system's base is immobile, the modeled base is fixed after deriving the equations. Six additional base degrees of freedom $\boldsymbol{q}_\mathrm{b} = [x, y, z, \alpha, \beta, \gamma]^\mathrm{T}$ are introduced, with α, β, γ being the Cardan angles for base orientation and x, y, z the base displacement. In total, this leads to the full system described by nine minimal coordinates $\boldsymbol{q}_\mathrm{f} = [\boldsymbol{q}_\mathrm{b}, \boldsymbol{q}_\mathrm{r}]^\mathrm{T}$. The kinematic modeling is finished by the mapping between the CC space and the Cartesian space, featuring the forward kinematics $\boldsymbol{f}_\mathrm{EE}(\boldsymbol{q}_\mathrm{f}) \in \mathbb{R}^3$ and the Jacobian $\boldsymbol{J}(\boldsymbol{q}_\mathrm{f}) = \partial \boldsymbol{f}_\mathrm{EE}(\boldsymbol{q}_\mathrm{f}) / \partial \boldsymbol{q}_\mathrm{f} \in \mathbb{R}^{3\times 9}$

$$\begin{aligned} \boldsymbol{x}_\mathrm{EE} &= \boldsymbol{f}_\mathrm{EE}(\boldsymbol{q}_\mathrm{f}) \\ \dot{\boldsymbol{x}}_\mathrm{EE} &= \boldsymbol{J}(\boldsymbol{q}_\mathrm{f})\ \dot{\boldsymbol{q}}_\mathrm{f}. \end{aligned} \tag{1}$$

For better readability, the dependencies aren't included in the formula in the remainder of the paper. Please note, as only one CC segment is considered and the base is fixed, only the Cartesian position is regarded. Considerations of the orientation can be omitted.

With the kinematics model, assumptions for the dynamics are made. The robot's mass is assumed as a point mass in the arc's center of gravity, the material behavior is considered as a linear model for stiffness and bending. The potential and kinetic energy of the point mass are derived and used in the Lagrangian equations of the second kind to obtain the corresponding model. The equations of motion for the floating-base system are derived as

$$\begin{bmatrix} \boldsymbol{M}_\mathrm{bb} & \boldsymbol{M}_\mathrm{br} \\ \boldsymbol{M}_\mathrm{rb} & \boldsymbol{M}_\mathrm{rr} \end{bmatrix} \begin{bmatrix} \ddot{\boldsymbol{q}}_\mathrm{b} \\ \ddot{\boldsymbol{q}}_\mathrm{r} \end{bmatrix} + \begin{bmatrix} \boldsymbol{b}_\mathrm{b} \\ \boldsymbol{b}_\mathrm{r} \end{bmatrix} = \begin{bmatrix} \boldsymbol{\tau}_\mathrm{react} \\ \boldsymbol{\tau}_\mathrm{act} \end{bmatrix}. \tag{2}$$

The inertia matrix is denoted as $\boldsymbol{M}(\boldsymbol{q}_\mathrm{f}) \in \mathbb{R}^{9\times 9}$ (parameter: robot's mass m). The Coriolis and centrifugal forces, the forces resulting from gravity and the forces resulting from the material model are summarized as the bias force vector $\boldsymbol{b}(\boldsymbol{q}_\mathrm{f}, \dot{\boldsymbol{q}}_\mathrm{f}) \in \mathbb{R}^9$ (parameters: m, bending stiffness/damping k_θ, d_θ, elongation stiffness/damping $k_{\delta L}, d_{\delta L}$, gravity constant g). In (2), the indices denote whether a component is associated with the floating base (b) or the deforming robotic system (r), with the sizes of the diagonal elements of the inertia matrix $\boldsymbol{M}_\mathrm{bb}(\boldsymbol{q}_\mathrm{f}) \in \mathbb{R}^{6\times 6}$ and $\boldsymbol{M}_\mathrm{rr}(\boldsymbol{q}_\mathrm{f}) \in \mathbb{R}^{3\times 3}$.

On the right-hand side of the equation, $\boldsymbol{\tau}_{\mathrm{react}}$ refers to the base reaction forces and torques, and the driving force vector $\boldsymbol{\tau}_{\mathrm{act}}(\boldsymbol{p}, \boldsymbol{q}_{\mathrm{f}})$ is computed by the deformation state and the input pressure $\boldsymbol{p}$. For the experimental study conducted to evaluate the proposed filter, ground truth data is needed. It is provided by a camera system, which requires a rigid body with markers at the tip of the SPA that is modeled as a point mass. Including the rotational inertia is possible, but with the experimental results, no significant effect of the rotational inertia was observed. The markers have to be considered with the marker inertia matrix $\boldsymbol{M}_{\mathrm{m}}(\boldsymbol{q}_{\mathrm{f}}) \in \mathbb{R}^{9\times 9}$ and the marker bias forces vector $\boldsymbol{b}_{\mathrm{m}}(\boldsymbol{q}_{\mathrm{f}}, \dot{\boldsymbol{q}}_{\mathrm{f}}) \in \mathbb{R}^{9}$ to ensure the correctness of the model.

$$\begin{aligned}\boldsymbol{M}_{\mathrm{m}} &= m_{\mathrm{m}}\, \boldsymbol{J}^{\mathrm{T}}\boldsymbol{J} \\ \boldsymbol{b}_{\mathrm{m}} &= m_{\mathrm{m}}\, \boldsymbol{J}^{\mathrm{T}}\left(\dot{\boldsymbol{J}}\dot{\boldsymbol{q}}_{\mathrm{f}} - [0,0,\mathrm{g}]^{\mathrm{T}}\right).\end{aligned} \tag{3}$$

In this case, the only additional parameter is the markers' mass m_{m}. For the experimental evaluation, the quantities in (3) are added to the corresponding quantities in (2); for readability, they aren't noted in the remainder of the paper.

B. State-Space Representation

A state-space representation of the model in (2), consisting of state and output equations, is needed inside the UKF. The first step is to fix the base by inserting the boundary condition $\ddot{\boldsymbol{q}}_{\mathrm{b}} = \dot{\boldsymbol{q}}_{\mathrm{b}} = \boldsymbol{q}_{\mathrm{b}} = \boldsymbol{0}$. This leads to the decoupled equations for the system's base-reaction forces and torques

$$\boldsymbol{M}_{\mathrm{br}}\,\ddot{\boldsymbol{q}}_{\mathrm{r}} + \boldsymbol{b}_{\mathrm{b}} = \boldsymbol{\tau}_{\mathrm{react}}, \tag{4}$$

and the actuation forces

$$\boldsymbol{M}_{\mathrm{rr}}\,\ddot{\boldsymbol{q}}_{\mathrm{r}} + \boldsymbol{b}_{\mathrm{r}} = \boldsymbol{\tau}_{\mathrm{act}}. \tag{5}$$

The state equation

$$\begin{bmatrix}\dot{\boldsymbol{q}}_{\mathrm{r}} \\ \ddot{\boldsymbol{q}}_{\mathrm{r}}\end{bmatrix} = \begin{bmatrix}\dot{\boldsymbol{q}}_{\mathrm{r}} \\ \boldsymbol{M}_{\mathrm{rr}}^{-1}\ (\boldsymbol{\tau}_{\mathrm{act}} - \boldsymbol{b}_{\mathrm{r}})\end{bmatrix} \tag{6}$$

is obtained by solving (5) for $\ddot{\boldsymbol{q}}_{\mathrm{r}}$ and defining the state concerning the CC model as $[\boldsymbol{q}_{\mathrm{r}}, \dot{\boldsymbol{q}}_{\mathrm{r}}]^{\mathrm{T}}$ The output equation is formed by inserting (6) into (4), giving

$$\boldsymbol{\tau}_{\mathrm{react}} = \boldsymbol{M}_{\mathrm{br}}\left(\boldsymbol{M}_{\mathrm{rr}}^{-1}\left(\boldsymbol{\tau}_{\mathrm{act}} - \boldsymbol{b}_{\mathrm{r}}\right)\right) + \boldsymbol{b}_{\mathrm{b}}. \tag{7}$$

This way, the measurable base-reaction forces and torques are part of the modeled state-space representation and can be used for state estimation.

C. Discretized Model for UKF

Kalman filters work with discrete-time models; therefore, (6) and (7) must be discretized. In empirical studies, stability issues occurred when simple explicit discretization methods were used. On the other hand, implicit or higher-order explicit methods make real-time capability more challenging. This is solved by utilizing the semi-implicit Euler method [27]. By updating the configuration states $\boldsymbol{q}_{\mathrm{r}}$, and then computing the velocity states $\dot{\boldsymbol{q}}_{\mathrm{r}}$ using the updated configuration, stability is achieved at larger time steps compared to discretization with an explicit Euler method.

The kinematic assumption of CC introduces a significant error compared to the deformation of the real system. The bending stiffness $k_\theta = x_{\mathrm{s}3}$ is joined to the state vector to mitigate this effect. By defining the full state vector as $\boldsymbol{x}_{\mathrm{s}} = [\boldsymbol{x}_{\mathrm{s}1}, \boldsymbol{x}_{\mathrm{s}2}, x_{\mathrm{s}3}]^{\mathrm{T}} = [\boldsymbol{q}_{\mathrm{r}}, \dot{\boldsymbol{q}}_{\mathrm{r}}, k_\theta]^{\mathrm{T}}$, the output as $\boldsymbol{y}_{\mathrm{s}} = \boldsymbol{\tau}_{\mathrm{react}}$, and the input as $\boldsymbol{u}_{\mathrm{s}} = \boldsymbol{p}$, the state-space representation is discretized with the state $\dot{\boldsymbol{x}}_{\mathrm{s}} = \boldsymbol{f}(\boldsymbol{x}_{\mathrm{s}}, \boldsymbol{u}_{\mathrm{s}})$ and output equation $\boldsymbol{y}_{\mathrm{s}} = \boldsymbol{h}(\boldsymbol{x}_{\mathrm{s}}, \boldsymbol{u}_{\mathrm{s}})$.

The semi-implicit Euler discretization of (6), including the bending stiffness, is

$$\begin{aligned}\boldsymbol{x}_{\mathrm{s},k+1} &= \boldsymbol{f}_{\mathrm{d}}(\boldsymbol{x}_{\mathrm{s},k}, \boldsymbol{u}_{\mathrm{s},k+1}) + \boldsymbol{w}_k \\ \begin{bmatrix}\boldsymbol{x}_{\mathrm{s}1,k+1} \\ \boldsymbol{x}_{\mathrm{s}2,k+1} \\ x_{\mathrm{s}3,k+1}\end{bmatrix} &= \begin{bmatrix}\boldsymbol{x}_{\mathrm{s}1,k} + \Delta t\, \boldsymbol{x}_{\mathrm{s}2,k} \\ \boldsymbol{x}_{\mathrm{s}2,k} + \Delta t\, \left(\boldsymbol{M}_{\mathrm{rr}}^{-1}\ (\boldsymbol{\tau}_{\mathrm{act}} - \boldsymbol{b}_{\mathrm{r}})\right) \\ x_{\mathrm{s}3,k}\end{bmatrix} + \boldsymbol{w}_k,\end{aligned} \tag{8}$$

with the quantities evaluated at the discrete time steps $\boldsymbol{M}_{\mathrm{rr}} = \boldsymbol{M}_{\mathrm{rr}}(\boldsymbol{x}_{\mathrm{s}1,k+1})$, $\boldsymbol{b}_{\mathrm{r}} = \boldsymbol{b}_{\mathrm{r}}(\boldsymbol{x}_{\mathrm{s}1,k+1}, \boldsymbol{x}_{\mathrm{s}2,k}, x_{\mathrm{s}3,k})$, $\boldsymbol{\tau}_{\mathrm{act}} := \boldsymbol{\tau}_{\mathrm{act}}(\boldsymbol{x}_{\mathrm{s}1,k+1}, \boldsymbol{u}_{\mathrm{s},k+1})$ and the time step size Δt. Process noise is included by $\boldsymbol{w}_k$. It is assumed to be normally distributed with the covariance matrix $\boldsymbol{Q} \in \mathbb{R}^{7\times 7}$. No dynamics for the stiffness is considered. Change of the parameter is only driven by process noise. This resembles the joint UKF setup in [28]. Analogously, the output equation is discretized by

$$\begin{aligned}\boldsymbol{y}_{\mathrm{s},k} &= \boldsymbol{h}(\boldsymbol{x}_{\mathrm{s},k}, \boldsymbol{u}_{\mathrm{s},k}) + \boldsymbol{v}_k \\ \boldsymbol{y}_{\mathrm{s},k} &= \boldsymbol{M}_{\mathrm{br}}\left(\boldsymbol{M}_{\mathrm{rr}}^{-1}\left(\boldsymbol{\tau}_{\mathrm{act}} - \boldsymbol{b}_{\mathrm{r}}\right)\right) + \boldsymbol{b}_{\mathrm{b}} + \boldsymbol{v}_k,\end{aligned} \tag{9}$$

with $\boldsymbol{M}_{\mathrm{br}} = \boldsymbol{M}_{\mathrm{br}}(\boldsymbol{x}_{\mathrm{s}1,k})$, $\boldsymbol{M}_{\mathrm{rr}} = \boldsymbol{M}_{\mathrm{rr}}(\boldsymbol{x}_{\mathrm{s}1,k})$, $\boldsymbol{b}_{\mathrm{b}} = \boldsymbol{b}_{\mathrm{b}}(\boldsymbol{x}_{\mathrm{s},k})$, $\boldsymbol{b}_{\mathrm{r}} = \boldsymbol{b}_{\mathrm{r}}(\boldsymbol{x}_{\mathrm{s},k})$, $\boldsymbol{\tau}_{\mathrm{act}} = \boldsymbol{\tau}_{\mathrm{act}}(\boldsymbol{x}_{\mathrm{s}1,k}, \boldsymbol{u}_{\mathrm{s},k})$ and the measurement noise $\boldsymbol{v}_k$, which is also assumed as a normal distribution with the covariance matrix $\boldsymbol{R} \in \mathbb{R}^{6\times 6}$. As a result of these modeling assumptions, the state vector to be estimated consists of seven entries: the CC parameters, their rate of change, and the bending stiffness.

The UKF uses the unscented transform by [29], [28]. It is performed by evaluating the state-space representation (8) and (9) at $2N+1$ sigma points with $N = \dim(\boldsymbol{x}_{\mathrm{s}}) = 7$. The state estimation consists of two steps. A prediction step by the derived model and a correction step by the acquired measurement. Further details on the UKF algorithm can be obtained from [28]. The required parameters are the time step size Δt, the covariance matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$, initial values for state estimation $\hat{\boldsymbol{x}}_{\mathrm{s},0}$ and error covariance $\boldsymbol{P}_0$, the scalar tuning parameters α_{UKF}, β_{UKF}, κ_{UKF}, the input pressure $\boldsymbol{u}_{\mathrm{s}}$ and the measured base-reaction forces and torques $\boldsymbol{y}_{\mathrm{s}}$.

III. Experiment

In the following, the experiments used to validate the proposed methods are explained and evaluated.

A. Setup

Experimental data is gathered on an SPA test bench depicted in Fig. 1. The SPA's base is mounted on the force-torque sensor (mini40, ATI Industrial Automation). Actuation is performed by letting air into the chambers. The actuation pressure lies between approximately 0 and $1\,\mathrm{bar}$ relative

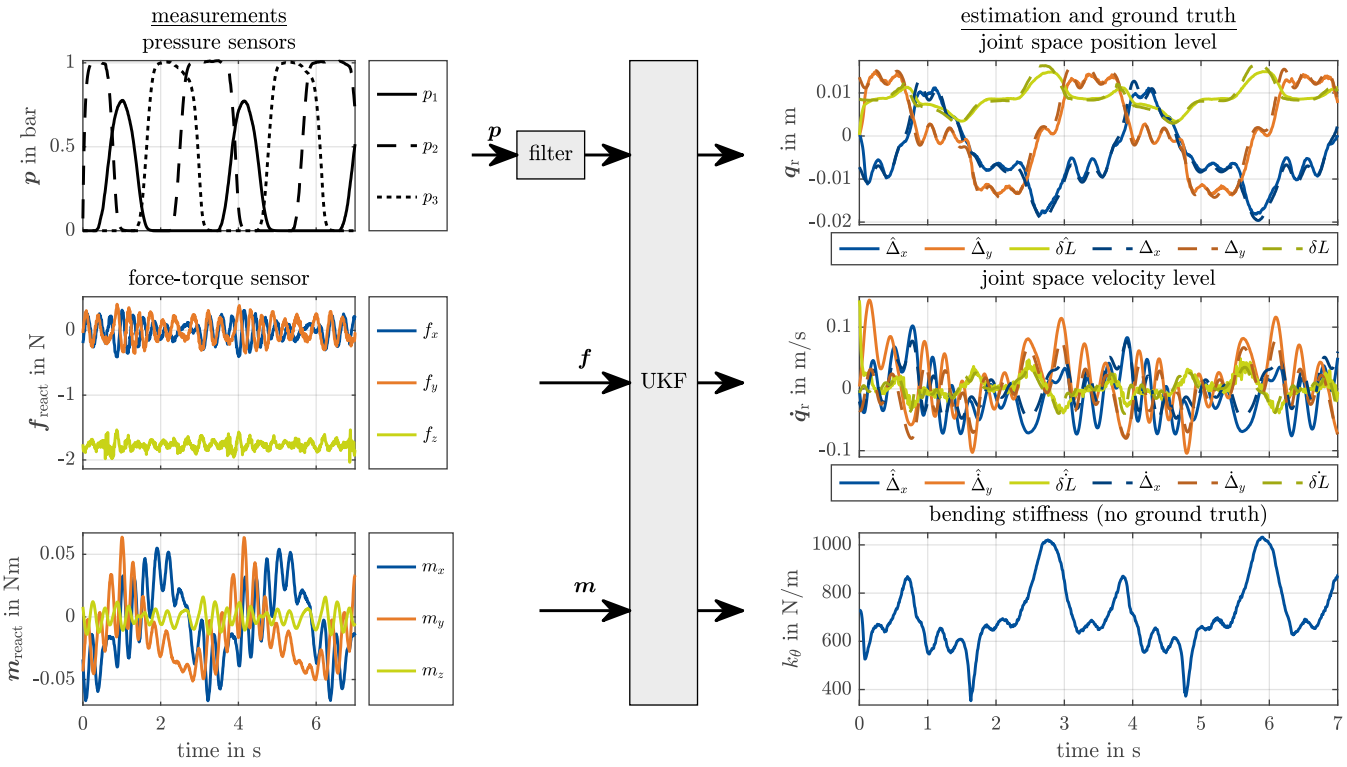

Fig. 2. Even though the initialization is not near the true initial value, the estimation converges. On the velocity data, the remaining error is greater compared to the positional data. For the bending stiffness, nonlinear behavior is observed. Inputs are shown on the left side.

to atmospheric pressure. It is measured directly before the pressure chambers (First Sensors, 142BC30A-PCB). The trajectory is generated by feedforward control of the solenoid valves. Thus, no pressure control is regarded. As ground truth for the state estimation, the actuator's tip Cartesian position is measured by external optical measurement (camera system, OptiTrack, Prime 17W). The CC parameters are obtained by inversion of (1) using Newton's method. Measurement is conducted by tracking the optical markers shown in the picture. The Cartesian velocity is obtained by applying a zero-phase moving average filter, which avoids a phase shift of the ground truth data, followed by numerical differentiation. On the other hand, the measured pressure signal is pre-filtered by a first-order Butterworth filter because it is a causal filter suitable for online applications in contrast to the zero-phase moving average filter. Since the UKF is supposed to run online, using a non-causal filter would falsify the results. Measured data is fed into the UKF implemented in MATLAB and compiled into a mex-file to test for real-time capability, using an INTEL i7 CPU at 2.8 GHz running UBUNTU 22.4 with the generic kernel. These specs are similar to the implemented real-time computer at our test bench [12]. A sample time of $\Delta t = 8\,\mathrm{ms}$ is used to ensure the stability of the UKF. The measurement data is obtained at the same frequency. The other UKF inputs are measurement noise covariance $\boldsymbol{R} = 1\mathrm{e}{-6}\ \mathrm{diag}\,(1,1,1,0.01,0.01,0.01)$, process noise covariance $\boldsymbol{Q} = 1\mathrm{e}{-9}\ \mathrm{diag}\,(1,1,1,0.1,0.1,0.1,1\mathrm{e}8)$, initial value for state estimation $\hat{\boldsymbol{x}}_{\mathrm{s},0} = [0,0,0,0,0,0,k_{\theta,0}]^{\mathrm{T}}$, initial value for error covariance $\boldsymbol{P}_0 = \boldsymbol{Q}$ and the scalar tuning parameters $\alpha_{\mathrm{UKF}} = 1$, $\beta_{\mathrm{UKF}} = 2$, $\kappa_{\mathrm{UKF}} = 0$. The input pressure and measured base reaction forces and torques can be seen on the left side of Fig. 2. With the parameters for the kinematics known from CAD, the parameters for the dynamics are experimentally identified, resembling the inner loop in [13], without identifying parameters for kinematics. The results can be seen in the first row of Table I.

B. Results

A periodic excitation of the system is chosen to test the proposed methods experimentally. Details of the trajectory can be seen in the supplementary video material. Data of 100 s of actuation is acquired, starting from the static straight configuration. This full data set is split into ten parts of 10 s so that the initial conditions of each part are different, while the initialization for the filter always stays the one stated above. This way, the convergence of the filter is tested. The estimation results, compared to the ground truth data of 7 s of the ninth part, are depicted on the right side of Fig. 2, with the motion of the whole part visualized in Cartesian space in Fig. 3. Cartesian estimation results of the tip position are obtained by evaluating (1) using the estimated CC parameters.

TABLE I
OFFLINE IDENTIFIED MODEL PARAMETERS

	r mm	L_0 mm	m_{m} kg	m kg	d_θ N s/m	$d_{\delta L}$ N s/m	$k_{\theta,0}$ N/m	$k_{\delta L}$ N/m
Exp	21	130	0.05	0.13	1.83	9.19	729	1015
Sim	21	130	0	0.16	0.15	6.23	505	1128

TABLE II

RESULTING ERRORS ON MEASUREMENT AND SIMULATION DATA

	Data	NRMSE in %							$\bar{e}_x$	max deflection	$\bar{e}_{\dot{x}}$	max velocity
		Δ_x	Δ_y	δL	$\dot{\Delta}_x$	$\dot{\Delta}_y$	$\dot{\delta} L$	**mean**	in mm	in mm	in $^{\text{mm}}/_{\text{s}}$	in $^{\text{mm}}/_{\text{s}}$
Exp	full	6.9	8.4	5.9	20.4	20.3	21.4	**13.9**	5.2	63.9	76.6	398.0
	best	6.9	8.7	6.3	17.3	21.6	21.5	**13.7**	5.2	63.5	69.9	398.0
	worst (Fig. 2)	6.6	8.5	5.7	25.9	31.0	23.1	**16.8**	5.0	63.7	91.8	364.3
Sim	full	4.3	4.6	6.4	18.4	26.0	12.7	**12.0**	3.1	57.4	57.5	275.7
	best	4.2	4.6	6.0	18.6	27.0	12.5	**11.5**	2.9	57.3	57.4	275.7
	worst (Fig. 5)	4.7	4.6	6.4	24.6	38.2	13.2	**15.3**	3.2	56.8	63.7	271.4

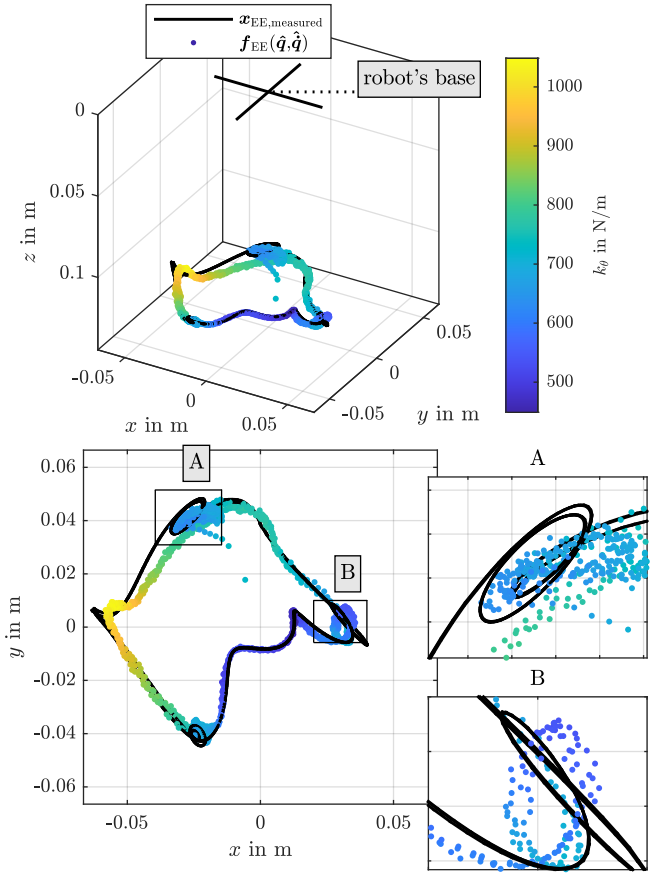

Fig. 3. 3D plot of the results mapped to the Cartesian space. The estimated stiffness is consistent over multiple periods. The observed stiffness is used to mitigate modeling errors affecting dynamics.

In both figures, the estimated bending stiffness is outlined. Because of the simplified model, a non-constant stiffness behavior is estimated by the filter. An example is given in Fig. 2. For the full data set and the first part, the initialized estimation fits the real starting state, but convergence was achieved even for the other parts, for which there is no agreement between initialization and the real starting state. In Table II, the resulting errors are shown. In order to calculate the error after convergence, the first approximately 0.1 s are excluded. The root-mean-square error, normalized to the maximum of the ground truth of each quantity (NRMSE) of all states, excluding the bending stiffness, is listed together with error measures in the Cartesian space. Resulting errors for three data sets, the full data set as well as the parts with

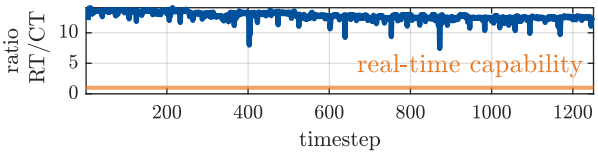

Fig. 4. Demonstration of real-time capacity.

the best (part 1) and worst (part 9) mean NRMSE (over the six computed NRMSE values per part) are visible. The filter estimates the state with an NRMSE of around 7% (5.7% to 8.7%) on the configuration and around 20% (17.3% to 25.9%) on the velocity data on various cutouts of 10 s length. As seen on the worst estimated data part, an outlier NRMSE of 31.0% is obtained. Regarding the Cartesian position data, a mean error of the tip position generated from the estimation $\bar{e}_x$ of approximately 5 mm at a maximal deflection of around 64 mm from the straight configuration is observed. Compared to the Cartesian position error, Cartesian velocity errors are larger, as was also indicated by the NRMSE. The mean error on the velocity $\bar{e}_{\dot{x}}$ ranges from 69.9 $^{\text{mm}}/_{\text{s}}$ to 91.8 $^{\text{mm}}/_{\text{s}}$, with a maximum velocity just short of 400 $^{\text{mm}}/_{\text{s}}$.

The motivation for utilizing CC kinematics for modeling is the possibility of real-time state estimation. This is tested by running the compiled UKF as *Matlab executable* (mex) and measuring its computation time (CT) in MATLAB. The real-time (RT) is equal to the sample time Δt of Section III-A. The results are shown in Fig. 4, a value over one indicates the computation of the time step in real-time. Real time state estimation is possible since the ratio of sample time to computation time is around 13 to 14 after the filter is initialized.

IV. SIMULATION

A downside of the presented experimental evaluation is the necessity of attaching a rigid body to the robot's tip to gain ground-truth information. The aim of the presented method is to eliminate the need to add rigid bodies to the soft system.

A. Setup

A Cosserat rod simulation is utilized to test the filter without an additional rigid body at the robot's tip. This model approximates the real SPA more accurately than CC models, but the applied numerical method comes at higher computational costs, making it infeasible for application inside the UKF. The Cosserat rod simulation was optimized to fit the real SPA as closely as possible. For more information regarding the simulation environment, refer to [30].

For actuation, the same pressure signal as for the experimental validation is used. The simulation is conducted with $\Delta t = 5\,\mathrm{ms}$. The same time step size is used in the UKF. The output of the simulation consists of the pose and velocity of the center-line points of the SPA as well as the base reaction forces and torques. With the reduced mass, stability conditions for the CC model change. As less mass is present in the system, a higher sample frequency is necessary for the stability of the numerical integration. Real-time computation remains possible with the smaller time step size. The model parameters are shown at the bottom of Table I. Except for the system noise covariance, which is tuned heuristically to $\boldsymbol{Q} = 1\mathrm{e}{-10}\ \mathrm{diag}\,(1, 1, 1, 0.1, 0.1, 0.1, 1\mathrm{e}9)$, the UKF parameters remain unchanged compared to the experimental validation. Because a simulation generated the base reaction forces and torques, the added measurement noise is perfectly known, and the measurement noise covariance matrix fits the data exactly.

B. Results

Parallel to the experimental results, the simulation results are split into ten parts of $10\,\mathrm{s}$. Seven seconds of state estimation results of part nine can be seen in Fig. 5, where convergence is observed. Contrary to the experimental results, no convergence was achieved without additional tuning of the $\boldsymbol{Q}$ entries in two cases: part 3 and part 8. As these data sets need special tuning, they are excluded from the error comparison. Errors for the other data sets are given at the bottom of Table II. As for the experimental results, errors of three data sets are shown: the full data set, the part with the best mean NRMSE (part 1), and the worst mean NRMSE (part 10). Observed errors are smaller compared to the experiment case, with NRMSE of 4.3% to 6.4% for the configuration and 12.5% to 27.0%, with one outlier of 38.2% at the worst result for the velocity estimation. Improvement can also be seen in the Cartesian results. Mean errors on the position range between $2.9\,\mathrm{mm}$ and $3.2\,\mathrm{mm}$ with a maximal deflection of approximately $58\,\mathrm{mm}$. For the velocity, the mean error lies between $57.5\,\mathrm{mm/s}$ and $63.7\,\mathrm{mm/s}$, with the maximum velocity of around $275\,\mathrm{mm/s}$. This finding is consistent with the statement from [31] that with a load on the robot, the error introduced by the CC kinematics becomes larger.

V. Conclusion

Estimation of the system's states is achieved by evaluating the input combined with force-torque sensor readings at the robot's base without affecting the system's dynamics by introducing additional mass. The results regarding the accuracy and frequency of state estimation indicate a high potential for control experiments of free-movement tasks. For evaluation, the proposed filtering technique was utilized on data gathered by a simulation employing the geometrically exact Cosserat rod model with known noise of the measurements and on data gathered on the real system. Similar resulting observation errors were obtained. The proposed method allows real-time state estimation using a simple model of CC dynamics,

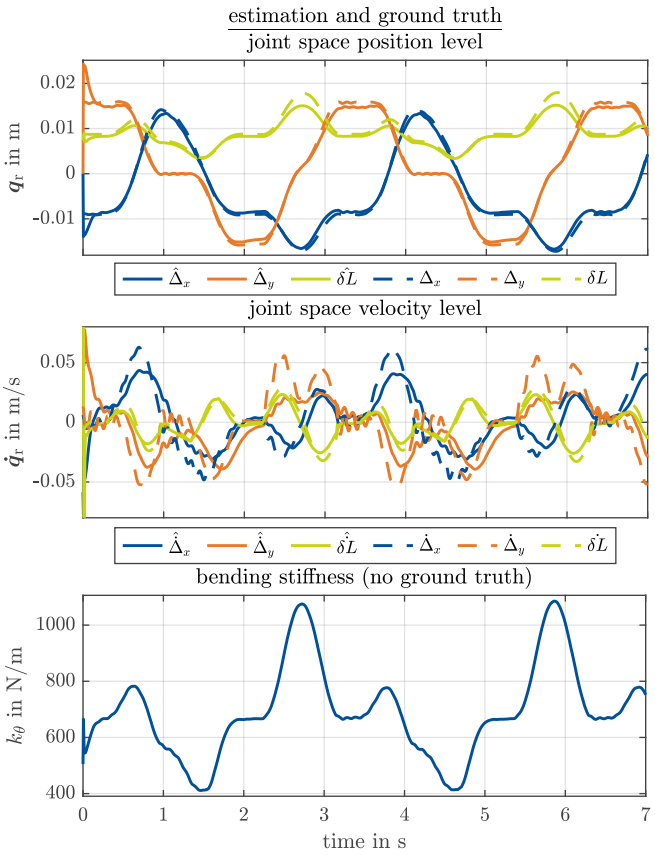

Fig. 5. Worst estimation results showing convergence on simulation data. As the same input was used, the results match the experimental study.

including the base-reaction forces and torques with the well-known UKF, by incorporating the bending stiffness into the estimation scheme. The implemented material law within the Cosserat rod model is a linear relationship between local moments and rotational strain quantities. Therefore, we conclude that the estimated nonconstant bending stiffness parameter has no further meaning but to compensate for the strong kinematic CC assumption. Including further parameters, e.g. damping or mass, is possible in the same manner but would require a online sensitivity analysis to avoid parameter drift due to the lack of parameter excitation. Estimation on the velocity level is less satisfactory than on the position level. If needed, velocity data can be obtained by differentiating the position data, but it is important to point out that in the dynamic case, velocity data is always part of the model. Therefore, the estimation of velocity comes without additional computational costs. Optimization of covariance matrices could lead to improved velocity estimations. Future research will focus on comparing the presented results to results from simpler filters, like the EKF, and utilizing more complex models like variable curvature or rod models.

References

[1] C. Della Santina, C. Duriez, and D. Rus, "Model-based control of soft robots: A survey of the state of the art and open challenges," *IEEE Control Systems*, vol. 43, no. 3, pp. 30–65, 2023.

[2] S. Ibrahim, J. C. Krause, A. Olbrich, and A. Raatz, "Modeling and reconstruction of state variables for low-level control of soft pneumatic actuators," *Frontiers in robotics and AI*, vol. 8, p. 557830, 2021.

[3] E. R. Rosi, M. Stolzle, F. Solari, and C. Della Santina, "Sensing soft robots' shape with cameras: an investigation on kinematics-aware SLAM," in *2022 IEEE 5th International Conference on Soft Robotics (RoboSoft)*. IEEE, 2022, pp. 795–801.

[4] A. Hassan, F. Aljaber, H. Godaba, I. Vitanov, and K. Althoefer, "Soft multi-point waveguide sensor for proprioception and extereoception in inflatable fingers," in *2021 IEEE 6th International Forum on Research and Technology for Society and Industry (RTSI)*. IEEE, 2021, pp. 574–579.

[5] F. Aljaber, A. Hassan, I. Vitanov, and K. Althoefer, "Curvature and contact sensing with optical waveguides for soft silicone pneumatic actuator," in *2022 IEEE 5th International Conference on Soft Robotics (RoboSoft)*. IEEE, 2022, pp. 859–864.

[6] Y. Toshimitsu, K. W. Wong, T. Buchner, and R. Katzschmann, "SoPrA: Fabrication & dynamical modeling of a scalable soft continuum robotic arm with integrated proprioceptive sensing," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 653–660.

[7] H. Wang, M. Totaro, and L. Beccai, "Toward perceptive soft robots: Progress and challenges," *Advanced science (Weinheim, Baden-Wurttemberg, Germany)*, vol. 5, no. 9, p. 1800541, 2018.

[8] L. Scimeca, J. Hughes, P. Maiolino, and F. Iida, "Model-free soft-structure reconstruction for proprioception using tactile arrays," *IEEE Robotics and Automation Letters*, vol. 4, no. 3, pp. 2479–2484, 2019.

[9] S. Sadati, A. Shiva, N. Herzig, C. D. Rucker, H. Hauser, I. D. Walker, C. Bergeles, K. Althoefer, and T. Nanayakkara, "Stiffness imaging with a continuum appendage: Real-time shape and tip force estimation from base load readings," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2824–2831, 2020.

[10] H. Donat, J. Gu, and J. J. Steil, "Real-time shape estimation for concentric tube continuum robots with a single force/torque sensor," *Frontiers in robotics and AI*, vol. 8, p. 734033, 2021.

[11] R. Takano, H. Mochiyama, and N. Takesue, "Real-time shape estimation of Kirchhoff elastic rod based on force/torque sensor," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2017, pp. 2508–2515.

[12] M. Bartholdt, M. Wiese, M. Schappler, S. Spindeldreier, and A. Raatz, "A parameter identification method for static Cosserat rod models: Application to soft material actuators with exteroceptive sensors," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2021, pp. 624–631.

[13] M. Mehl, M. Bartholdt, and M. Schappler, "Dynamic modeling of soft-material actuators combining constant curvature kinematics and floating-base approach," in *2022 IEEE 5th International Conference on Soft Robotics (RoboSoft)*. IEEE, 2022, pp. 1–8.

[14] G. Ponraj and H. Ren, "Sensor fusion of leap motion controller and flex sensors using Kalman filter for human finger tracking," *IEEE Sensors Journal*, vol. 18, no. 5, pp. 2042–2049, 2018.

[15] K. Stewart, Z. Qiao, and W. Zhang, "State estimation and control with a robust extended Kalman filter for a fabric soft robot," *IFAC-PapersOnLine*, vol. 55, no. 37, pp. 25–30, 2022.

[16] D. Lunni, G. Giordano, E. Sinibaldi, M. Cianchetti, and B. Mazzolai, "Shape estimation based on Kalman filtering: Towards fully soft proprioception," in *2018 IEEE International Conference on Soft Robotics (RoboSoft)*. IEEE, 2018, pp. 541–546.

[17] J. Y. Loo, C. P. Tan, and S. G. Nurzaman, "H-infinity based extended Kalman filter for state estimation in highly non-linear soft robotic system," in *2019 American Control Conference (ACC)*. IEEE, 2019, pp. 5154–5160.

[18] J. Y. Loo, Z. Y. Ding, V. M. Baskaran, S. G. Nurzaman, and C. P. Tan, "Robust multimodal indirect sensing for soft robots via neural network-aided filter-based estimation," *Soft robotics*, vol. 9, no. 3, pp. 591–612, 2022.

[19] M. S. Xavier, A. J. Fleming, and Y. K. Yong, "Nonlinear estimation and control of bending soft pneumatic actuators using feedback linearization and UKF," *IEEE/ASME Transactions on Mechatronics*, vol. 27, no. 4, pp. 1919–1927, 2022.

[20] G. Zhang and L. Wang, "Shape estimation of continuum robots via modal parameterization and dual extended Kalman filter." [Online]. Available: http://arxiv.org/pdf/2210.08427v1

[21] L. Zeng, S. Sadati, and C. Bergeles, "Koopman operator-based extended Kalman filter for Cosserat rod wrench estimation," in *2023 International Symposium on Medical Robotics (ISMR)*. IEEE, 2023, pp. 1–7.

[22] S. Lilge, T. D. Barfoot, and J. Burgner-Kahrs, "Continuum robot state estimation using Gaussian process regression on SE(3)," *The International Journal of Robotics Research*, vol. 41, no. 13-14, pp. 1099–1120, 2022.

[23] T. Zheng, Q. Han, and H. Lin, "Full state estimation of continuum robots from tip velocities: A Cosserat-theoretic boundary observer." [Online]. Available: http://arxiv.org/pdf/2303.06130v3

[24] R. Szasz, M. Allenspach, M. Han, M. Tognon, and R. K. Katzschmann, "Modeling and control of an omnidirectional micro aerial vehicle equipped with a soft robotic arm," in *2022 IEEE 5th International Conference on Soft Robotics (RoboSoft)*. IEEE, 2022, pp. 01–08.

[25] C. Della Santina, A. Bicchi, and D. Rus, "On an improved state parametrization for soft robots with piecewise constant curvature and its use in model based control," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 1001–1008, 2020.

[26] I. S. Godage, R. Wirz, I. D. Walker, and R. J. Webster, "Accurate and efficient dynamics for variable-length continuum arms: A center of gravity approach," *Soft robotics*, vol. 2, no. 3, pp. 96–106, 2015.

[27] M. Gipser, *Systemdynamik und Simulation*. Wiesbaden: Vieweg+Teubner Verlag, 1999.

[28] E. A. Wan and R. van der Merwe, "The unscented Kalman filter for nonlinear estimation," in *Proceedings of the IEEE 2000 Adaptive Systems for Signal Processing, Communications, and Control Symposium (Cat. No.00EX373)*. IEEE, 2000, pp. 153–158.

[29] S. J. Julier and J. K. Uhlmann, "New extension of the Kalman filter to nonlinear systems," in *Signal Processing, Sensor Fusion, and Target Recognition VI*, ser. SPIE Proceedings, I. Kadar, Ed. SPIE, 1997, p. 182.

[30] M. Bartholdt, R. Berthold, and M. Schappler, "Towards a modular framework for visco-hyperelastic simulations of soft material manipulators with well-parameterised material," in *2023 IEEE International Conference on Soft Robotics (RoboSoft)*. IEEE, 2023, pp. 1–8.

[31] F. Stella, Q. Guan, C. Della Santina, and J. Hughes, "Piecewise affine curvature model: a reduced-order model for soft robot-environment interaction beyond PCC," in *2023 IEEE International Conference on Soft Robotics (RoboSoft)*. IEEE, 2023, pp. 1–7.