Preprints of the 21st IFAC World Congress (Virtual)
Berlin, Germany, July 12-17, 2020

Disturbance Observer Based Control for Quasi Continuum Manipulators

Daniel Müller * **Carina Veil** * **Oliver Sawodny** *

* *Institute for System Dynamics, Waldburgstraße 17/19, 70569 Stuttgart, E-Mail: daniel.mueller@isys.uni-stuttgart.de*

Abstract: Nowadays, robots are an essential part of modern production lines, usually working in a designated area since they can pose a threat to human workers. The so-called soft robots constitute a human-friendly alternative to classic industrial robots, even allowing for human-machine collaboration. This is possible due to their soft and therefore inherent safe structure. In this paper we consider quasi continuum manipulators (QCMs), a special kind of soft robots. Their dynamic behavior is affected by friction as well as their soft materials. Dynamical models are thus hard to identify, suffering from imperfections and uncertainties. To overcome these flaws we propose a disturbance observer (DOB) based controller using an extended Kalman filter (EKF). We show superior performance on a real robot compared to an existing benchmark concept based on a PID-like controller. The generalization of this approach is demonstrated by implementing our method on two QCMs with different kinematics.

Keywords: Disturbance Observer, Continuum Manipulator, Extended Kalman Filter.

1. INTRODUCTION

In the developing area of robots designed for interaction with humans, rigid structures are not the first choice anymore. Incompliant structures include a high risk of injury, which is why current research tends to a domain called 'soft robotics'. Early approaches in this domain were to use compliant joints within rigid-link robots, either based on variable stiffness, on compliance or impedance control (Albu-Schaffer et al. (2008)). Nowadays, the trend is towards bioinspired continuum manipulators (CM). The CMs are made of fully elastic structures instead of rigid links (Laschi et al. (2016)). In this paper we consider quasi continuum manipulators (QCMs) which were first described in our previous work (Müller et al. (2020)). They consist of pneumatic bellows, supported by a joint chain that allows quasi continuous bending – motivating the notion quasi continuum manipulator. Finding a suitable model which can be calculated online is a difficult task for these manipulators. Furthermore, the model identification is nontrivial due to the friction and material properties changing over time. This leads to unavoidable and notable imperfections in the dynamical model. Thus, controlling the continuum manipulator (CM) proves to be a difficult problem (Polygerinos et al. (2017)).

1.1 Literature Review and Previous Work

The review paper by Webster and Jones (2010) gives a good introduction on CMs and their kinematic models. Over the last couple of years, the constant curvature approach has emerged as the state of the art technique for modeling CMs.
Most of the existing controllers for CMs are based on this kinematic model. Thuruthel et al. (2018) gives an overview of the various approaches for controlling CMs. It is stated that model-based dynamical controllers are still in their nascent stage, possibly due to the hard derivation of dynamical models and their uncertainties. According to Thuruthel et al. (2018) current research trends go towards learning-based methods, thus avoiding tedious modeling. Nevertheless, model-based dynamical controllers find wide application in the industry, therefore the authors think that for an application outside of laboratories, this might be the way to go.

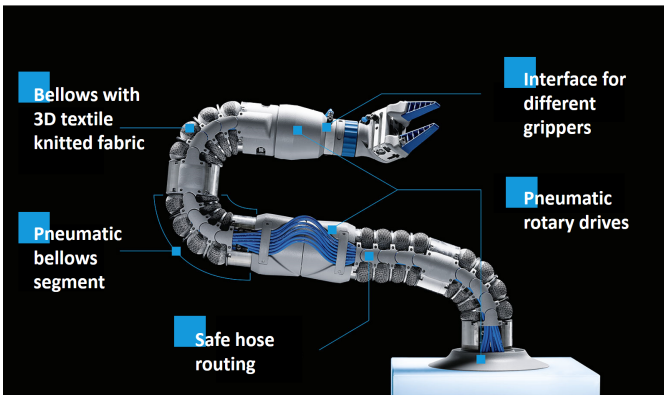

Fig. 1. Bionic Soft Arm. Displayed is the latest quasi continuum manipulator with seven degrees of freedom. Source: Festo

QCMs are not the first systems which suffer from unmodeled effects and disturbances, nearly all dynamical systems do. A common approach to deal with these effects is to simply use an integrator in the controller, as it has been done in our previous work (Müller et al. (2020)). However, this might lead to problems, especially since the behavior of the integrator during dynamical transitions is often

⋆ The authors gratefully acknowledge support of this work by the German Research Foundation (DFG) under grant SA 847/20-1.

poor. To increase performance, we implemented a gain scheduling scheme for the proposed integral controller. Thus, a PID-like controller was the result of Müller et al. (2020).

An alternative to an integrator is a disturbance observer (DOB). The main idea is to transfer the concept of conventional state observers, which reconstruct missing state information by knowledge of input and output to disturbances (Luenberger (1964)). This idea first arose in 1983 for torque-speed regulation of a DC motor (Ohishi (1983)) and has been widely extended since. The implementation of a so-called disturbance observer results in being able to estimate and compensate unknown disturbances. Once the compensation is done, regular control strategies can be applied in a cascaded fashion. The utilization of a DOB for control tasks has proven its worth in many different applications as mechatronics, chemical and aerospace systems and can be further extended (Chen et al. (2016)). Taking uncertainties in modeling or unmodeled effects, especially friction, as disturbances, one can achieve more robustness when using a DOB. The difficulty of the design of a robust controller is shifted to the design of the DOB.

An overview over DOB-based control is given in Chen et al. (2016). It shows that various kinds of disturbance and uncertainty estimation and attenuation exist, such as extended state observer or active disturbance rejection control. Successful implementations for rigid robotic manipulators are presented in Chen et al. (2000).

1.2 Contribution

The estimation and attenuation of disturbances and, more importantly uncertainties, in modeling is crucial in the domain of soft robotics, since the exact modeling of compliant structures proves difficult. In this paper, we address these issues by proposing a DOB-based control structure using an EKF which is successfully implemented on two QCMs. We further show that our approach has superior performance compared to a previously developed benchmark concept which uses a PID-like controller. To the best knowledge of the authors this has not been done for CMs.

The remainder of this paper is organized as follows. In Section 2 we state our problem formally. Afterwards, we present our solution in Section 3. Experimental results for two real QCMs are shown in Section 4 and are discussed in Section 5. Section 6 concludes the paper and states further research directions.

2. PROBLEM FORMULATION

In the following we state the problem addressed in this paper in a formal way. The original idea of the dynamical model is presented in Falkenhahn et al. (2014) for CMs. An adaption for QCMs with n degrees of freedom is given in (Müller et al. (2020)), which yields the mechanical model

$$\boldsymbol{M}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}} + \boldsymbol{N}(\boldsymbol{q}) = \boldsymbol{\tau} + \boldsymbol{\tau}_{\text{ext}}, \tag{1}$$

where $\boldsymbol{M}(\boldsymbol{q}) \in \mathbb{R}^{n\times n}$ is the inertia matrix, $\boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}}) \in \mathbb{R}^{n\times n}$ describes the Coriolis terms and $\boldsymbol{N}(\boldsymbol{q}) \in \mathbb{R}^n$ denotes the gravitational forces. The generalized coordinates are denoted by $\boldsymbol{q} \in \mathbb{R}^n$ and the generalized momentum, which is also the input for the mechanical plant, is expressed by $\boldsymbol{\tau} \in \mathbb{R}^n$. External disturbances such as friction and other effects are summed up in $\boldsymbol{\tau}_{\text{ext}} \in \mathbb{R}^n$.

Since one of the considered robots has rotational actuators as well as continuum actuators, $\boldsymbol{q}$ either denotes a rotational angle or a curvature of a continuum actuator. Furthermore, we assume that we can measure $\boldsymbol{q}$ directly, i.e. $\boldsymbol{y} = \boldsymbol{q}$. In Falkenhahn et al. (2015) it was shown that for low values of $\dot{\boldsymbol{q}}$ the Coriolis term $\boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})$ only has a small effect on the total energy of the bionic handling assistant. Based on these findings and the results of Müller et al. (2020), we conclude that this term can be neglected for the QCMs as well.

We further assume that model-plant mismatch is additive, i.e.

$$\boldsymbol{M}(\boldsymbol{q}) = \hat{\boldsymbol{M}}(\boldsymbol{q}) + \boldsymbol{\Delta}\hat{\boldsymbol{M}}(\boldsymbol{q}), \tag{2}$$

where $\boldsymbol{M}(\boldsymbol{q})$ is the real, $\hat{\boldsymbol{M}}(\boldsymbol{q})$ the modeled and $\boldsymbol{\Delta}\hat{\boldsymbol{M}}(\boldsymbol{q})$ the error inertia matrix. The same is assumed for the gravitational forces, which yields

$$\boldsymbol{N}(\boldsymbol{q}) = \hat{\boldsymbol{N}}(\boldsymbol{q}) + \boldsymbol{\Delta}\hat{\boldsymbol{N}}(\boldsymbol{q}), \tag{3}$$

where the notation is respectively the same as for (2).

As already stated in Section 1, the dynamic behavior of CMs is influenced by a wide variety of effects which are hard to capture by a model. These effects are summarized by the input disturbance $\boldsymbol{\xi} \in \mathbb{R}^n$. This yields the system considered in this paper, i.e.

$$\hat{\boldsymbol{M}}(\boldsymbol{q})\ddot{\boldsymbol{q}} + \hat{\boldsymbol{N}}(\boldsymbol{q}) = \boldsymbol{\tau} + \boldsymbol{\xi}, \tag{4}$$

with

$$\boldsymbol{\xi} = \boldsymbol{\tau}_{\text{ext}} - \boldsymbol{\Delta}\hat{\boldsymbol{M}}(\boldsymbol{q})\ddot{\boldsymbol{q}} - \boldsymbol{\Delta}\hat{\boldsymbol{N}}(\boldsymbol{q}) - \boldsymbol{C}(\boldsymbol{q},\dot{\boldsymbol{q}})\dot{\boldsymbol{q}}. \tag{5}$$

We can now state our problem in two parts.

Problem 1. Design an observer where its estimation $\hat{\boldsymbol{\xi}}$ converges to $\boldsymbol{\xi}$.

Problem 2. Control the system (4) using $\hat{\boldsymbol{\xi}}$.

3. SOLUTION APPROACH

The solution approach can be divided into two parts. First, we show observability of the disturbance $\boldsymbol{\xi}$ and estimate it using an EKF. In a second step, we propose a control structure using the estimation $\hat{\boldsymbol{\xi}}$ of $\boldsymbol{\xi}$ to impose the nominal behavior on the plant.

3.1 Disturbance Observer

In preparation for the disturbance observer (DOB), the dynamical model is extended by a disturbance state $\hat{\boldsymbol{\xi}}$. Since there is no information on how the disturbance state $\boldsymbol{\xi}$ changes over time it is assumed that its estimate $\hat{\boldsymbol{\xi}}$ has no momentum. This yields the dynamics of the extended state $\hat{\boldsymbol{x}} \in \mathbb{R}^{3n}$ as

$$\dot{\hat{\boldsymbol{x}}} = \begin{pmatrix} \dot{\hat{\boldsymbol{q}}} \\ \ddot{\hat{\boldsymbol{q}}} \\ \dot{\hat{\boldsymbol{\xi}}} \end{pmatrix} = \begin{pmatrix} \hat{\boldsymbol{q}} \\ \hat{\boldsymbol{M}}^{-1}(\hat{\boldsymbol{q}})(\boldsymbol{\tau} + \hat{\boldsymbol{\xi}} - \hat{\boldsymbol{N}}(\hat{\boldsymbol{q}})) \\ \boldsymbol{0} \end{pmatrix} = \boldsymbol{f}(\hat{\boldsymbol{x}}, \boldsymbol{\tau}), \tag{6}$$

where $\hat{\boldsymbol{q}}$, $\dot{\hat{\boldsymbol{q}}}$ and $\ddot{\hat{\boldsymbol{q}}}$ are estimates of the generalized coordinates, their velocities and accelerations respectively. The measurement equation is expressed by

$$\boldsymbol{y} = \boldsymbol{g}(\boldsymbol{x}) = (\, \boldsymbol{I}\ \boldsymbol{0}\ \boldsymbol{0} \,)\, \boldsymbol{x}. \tag{7}$$

Calculating the derivative of the right hand side of (6) yields

$$\frac{\partial \boldsymbol{f}(\hat{\boldsymbol{x}}, \boldsymbol{\tau})}{\partial \hat{\boldsymbol{x}}} = \begin{pmatrix} \boldsymbol{0} & \boldsymbol{I} & \boldsymbol{0} \\ \hat{\boldsymbol{p}}(\hat{\boldsymbol{x}}, \boldsymbol{\tau}) & \boldsymbol{0} & \hat{\boldsymbol{M}}^{-1}(\hat{\boldsymbol{q}}) \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{0} \end{pmatrix}, \tag{8}$$

where

$$\hat{\boldsymbol{p}}(\hat{\boldsymbol{x}}, \boldsymbol{\tau}) = \frac{\partial \hat{\boldsymbol{M}}^{-1}(\hat{\boldsymbol{q}})}{\partial \hat{\boldsymbol{q}}}(\boldsymbol{\tau} + \hat{\boldsymbol{\xi}} - \hat{\boldsymbol{N}}(\hat{\boldsymbol{q}})) - \hat{\boldsymbol{M}}^{-1}(\hat{\boldsymbol{q}}) \frac{\partial \hat{\boldsymbol{N}}(\hat{\boldsymbol{q}})}{\partial \hat{\boldsymbol{q}}} \tag{9}$$

and $\boldsymbol{I} \in \mathbb{R}^{n \times n}$ denotes the identity matrix. Note that $\frac{\partial}{\partial \hat{\boldsymbol{q}}} \hat{\boldsymbol{M}}^{-1}(\hat{\boldsymbol{q}})$ can be easily obtained using the chain rule, i.e.

$$\frac{\partial \hat{\boldsymbol{M}}^{-1}(\hat{\boldsymbol{q}})}{\partial \hat{\boldsymbol{q}}} = \hat{\boldsymbol{M}}^{-1}(\hat{\boldsymbol{q}}) \frac{\partial \hat{\boldsymbol{M}}(\hat{\boldsymbol{q}})}{\partial \hat{\boldsymbol{q}}} \hat{\boldsymbol{M}}^{-1}(\hat{\boldsymbol{q}}). \tag{10}$$

Observability The condition for observability is given by

$$\operatorname{rank}\left(\boldsymbol{Q}_{\mathrm{Obs}}(\boldsymbol{x})\right) = \operatorname{rank} \begin{pmatrix} \frac{\partial L_{\boldsymbol{f}}^{0} \boldsymbol{g}(\boldsymbol{x})}{\partial \boldsymbol{x}} \\ \vdots \\ \frac{\partial L_{\boldsymbol{f}}^{m-1} \boldsymbol{g}(\boldsymbol{x})}{\partial \boldsymbol{x}} \end{pmatrix} = 3n, \tag{11}$$

where $m = 3$ is the number of required derivatives. We can calculate $\boldsymbol{Q}_{\mathrm{Obs}}(\boldsymbol{x})$ for the original system using (7) and (8) with the original functions instead of its estimates. This yields

$$\boldsymbol{Q}_{\mathrm{Obs}}(\boldsymbol{x}) = \begin{pmatrix} \boldsymbol{I} & \boldsymbol{0} & \boldsymbol{0} \\ \boldsymbol{0} & \boldsymbol{I} & \boldsymbol{0} \\ \boldsymbol{p}(\boldsymbol{x}, \boldsymbol{\tau}) & \boldsymbol{0} & \boldsymbol{M}^{-1}(\boldsymbol{q}) \end{pmatrix}. \tag{12}$$

Since the inertia matrix $\boldsymbol{M}(\boldsymbol{q})$ has to be a positive definite matrix, its inverse exists and is also positive definite. Thus, $\boldsymbol{M}^{-1}(\boldsymbol{q})$ has full rank. It follows that $\boldsymbol{x}$ is observable and so is $\boldsymbol{\xi}$.

Extended Kalman Filter The EKF is a widely used technique for estimating states of nonlinear systems. It is an recursive algorithm that can be divided into a predictor and a corrector stage. We consider the discrete time implementation with the sample time Δt. Instead of restating the whole algorithm we want to point out that there exists plenty of tutorials on EKFs. A good introduction is given by Terejanu et al. (2008).
For sake of completeness, we state the necessary equations for implementing the EKF in the following: The process equation for the state $\hat{\boldsymbol{x}}_k$ in the kth step is obtained using

$$\hat{\boldsymbol{x}}_k = \boldsymbol{f}_{\mathrm{d}}(\hat{\boldsymbol{x}}_{k-1}, \boldsymbol{\tau}_{k-1}) = \hat{\boldsymbol{x}}_{k-1} + \Delta t \boldsymbol{f}(\hat{\boldsymbol{x}}_{k-1}, \boldsymbol{\tau}_{k-1}) \tag{13}$$

and its derivative is

$$\frac{\partial \boldsymbol{f}_{\mathrm{d}}}{\partial \hat{\boldsymbol{x}}_{k-1}} = \begin{pmatrix} \boldsymbol{I} & \Delta t \boldsymbol{I} & \boldsymbol{0} \\ \Delta t \hat{\boldsymbol{p}}(\hat{\boldsymbol{x}}_{k-1}, \boldsymbol{\tau}_{k-1}) & \boldsymbol{I} & \Delta t \boldsymbol{M}^{-1}(\hat{\boldsymbol{q}}_{k-1}) \\ \boldsymbol{0} & \boldsymbol{0} & \boldsymbol{I} \end{pmatrix}, \tag{14}$$

where we omitted the arguments of $\boldsymbol{f}_{\mathrm{d}}$ for reasons of space. The output function is simply the discrete version of (7), i.e. $\boldsymbol{y}_k = \boldsymbol{g}(\boldsymbol{x}_k) = (\, \boldsymbol{I} \; \boldsymbol{0} \; \boldsymbol{0} \,) \boldsymbol{x}_k$ and its derivative results in

$$\frac{\partial \boldsymbol{g}(\boldsymbol{x}_k)}{\partial \boldsymbol{x}_k} = (\, \boldsymbol{I} \; \boldsymbol{0} \; \boldsymbol{0} \,). \tag{15}$$

The covariance matrices of process noise $\boldsymbol{Q}$, measurement noise $\boldsymbol{R}$ and the initial covariance matrix $\boldsymbol{P}_0$ are empirically determined. Given that, the EKF is fully defined.

3.2 Control structure

The overall structure is illustrated in Figure 2. We divide the control law in multiple parts. The control action $\boldsymbol{\tau}_{\mathrm{ctrl}}$

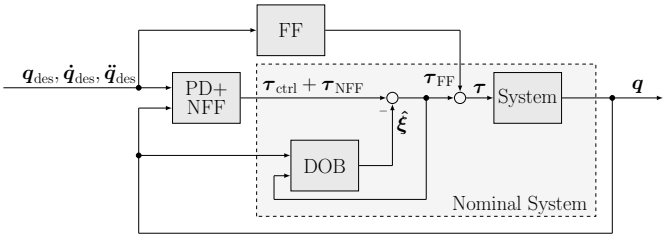

Fig. 2. Control Structure.

generated by the PD-Controller and the nominal feed forward (NFF) control $\boldsymbol{\tau}_{\mathrm{NFF}}$ is taken from Müller et al. (2020) with

$$\boldsymbol{\tau}_{\mathrm{ctrl}} = \hat{\boldsymbol{M}}(\hat{\boldsymbol{q}})(\boldsymbol{K}_{\mathrm{P}} \boldsymbol{e} + \boldsymbol{K}_{\mathrm{D}} \dot{\boldsymbol{e}}) \tag{16}$$

and

$$\boldsymbol{\tau}_{\mathrm{NFF}} = \hat{\boldsymbol{M}}(\hat{\boldsymbol{q}}) \ddot{\boldsymbol{q}}_{\mathrm{des}} + \hat{\boldsymbol{N}}(\hat{\boldsymbol{q}}) \tag{17}$$

where $\boldsymbol{e} = \boldsymbol{q}_{\mathrm{des}} - \hat{\boldsymbol{q}}$ is the error between the desired and the estimated general coordinates.
Furthermore, we have a feed forward (FF) part compensating friction and other effects which can be identified and compensated. Note that these effects are not part of the nominal model and might differ between various robots. An example for the manipulator displayed in Figure 1 is given in Müller et al. (2020). We collect these terms in $\boldsymbol{\tau}_{\mathrm{FF}}$. The input of the plant results in

$$\boldsymbol{\tau} = \boldsymbol{\tau}_{\mathrm{ctrl}} + \boldsymbol{\tau}_{\mathrm{NFF}} + \boldsymbol{\tau}_{\mathrm{FF}} - \hat{\boldsymbol{\xi}} \tag{18}$$

Note that according to Figure 2 the generalized moment input to the DOB is different from the input to the system (18). The input to the DOB lacks the extra $\boldsymbol{\tau}_{\mathrm{FF}}$ part and is given by $\boldsymbol{\tau}_{\mathrm{ctrl}} + \boldsymbol{\tau}_{\mathrm{NFF}} - \hat{\boldsymbol{\xi}}$.
If the control error is artificially maintained over a longer period of time, e.g. if the robot gets stuck, the disturbance state $\hat{\boldsymbol{\xi}}$ grows beynod limits. Thus, we modify the EKF by saturating the disturbance state, i.e. $\hat{\xi}_i \in [-c_{\max}, c_{\max}]$, where $\hat{\xi}_i$ is the i-th entry of $\hat{\boldsymbol{\xi}}$. The constant $c_{\max}$ limits the effects of the DOB on the plant. This value should be in the same order of magnitude as the control action $\boldsymbol{\tau}_{\mathrm{ctrl}}$. Note that this saturation is easily implemented and improves the safety of the overall system. It can especially prevent harm in scenarios where the error $\boldsymbol{e}$ is artificially maintained over a long period, e.g. someone holding back the manipulator.

4. EXPERIMENTAL RESULTS

We present two experiments. The first experiment shows a SCARA-like QCM, displayed in Figure 3. Its purpose is to give an indication on the behavior of the algorithm. For the second experiment we implemented the proposed algorithm on a seven degree of freedom QMC.
We used a DSpace rapid prototype system. The proposed control structure is running at 1 kHz using a DS1007 board. The MATLAB code is online available on Müller (2020).

4.1 SCARA-like QCM

The considered robot is shown in Figure 3. For a better understanding of the controller, we decided to have a series of step signals as an input $\boldsymbol{q}_{\mathrm{des}}$ with its derivatives set to

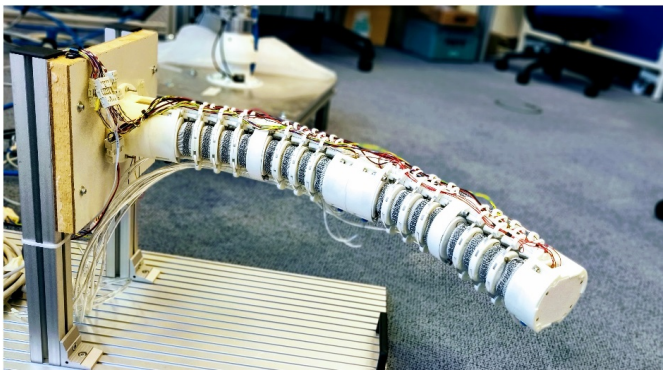

Fig. 3. SCARA-like QCM. The displayed robot has four degrees of freedom and can only move in a plane.

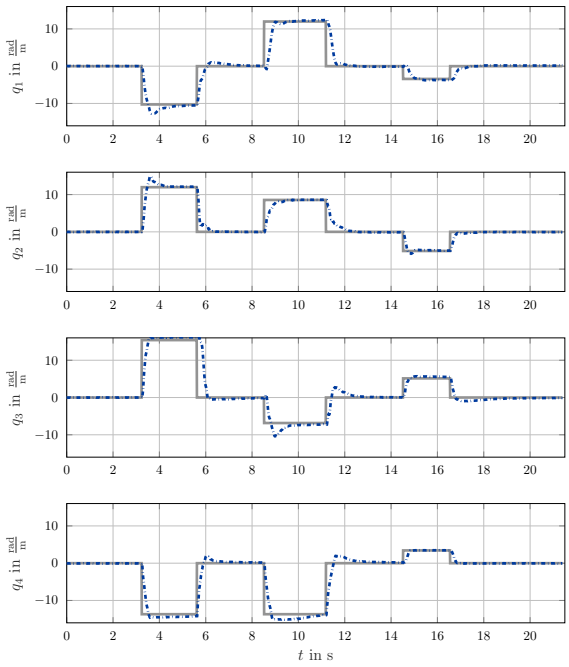

Fig. 4. Experimental data for the SCARA-like QCM. Depicted is the step response of the proposed controller. Reference signal: —. DOB-based controller: - · - · - .

zero. Although it is self evident that the system is unable to follow the input signal, since it has no direct feed through, the behavior of the overall system is indicated by the step response.

Figure 4 shows the desired input and output of its generalized coordinates. These coordinates correspond to the curvatures of the four segments, beginning with q_1 at the base of the robot and ending with q_4 right before the tool center point. For a better visualisation we also uploaded videos online (Müller (2019)).

4.2 QCM with Seven Degrees of Freedom

We consider the Bionic Soft Arm which was originally developed by Festo and is shown in Figure 1. It has four continuum actuators and three rotational swivel drives, thus resulting in a QCM with seven degrees of freedom. We let the robot follow a given trajectory and compare its performance with the previous proposed PID-like controller given in Müller et al. (2020). The results are separated

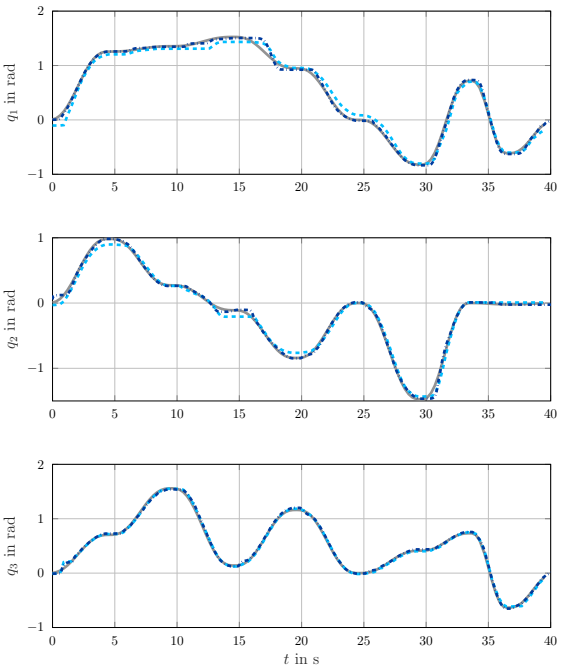

Fig. 5. Experiment data for the Bionic Soft Arm. Swivel drives following a given trajectory. Reference signal: —. DOB-based controller: - · - · - . PID-like controller: - - - .

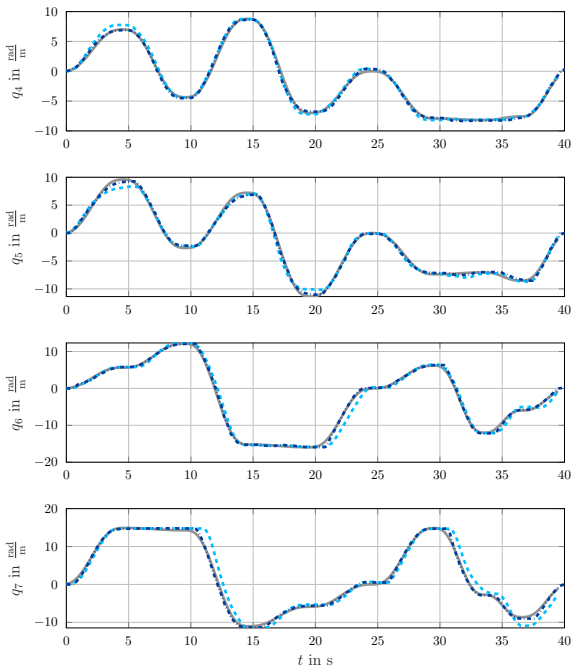

Fig. 6. Experiment data for the Bionic Soft Arm. Continuum actuators following a given trajectory. Reference signal: —. DOB-based controller: - · - · - . PID-like controller: - - - .

in rotational swivel drives in Figure 5 and continuum actuators in Figure 6. The separation was made because both drive concepts have different scales. A visualisation of the trajectory of both control concepts is uploaded as a video and can be found on Müller (2019).

We compute the mean absolute error of all swivel drives e_{SD} and all continuum actuators e_{CA}, i.e.

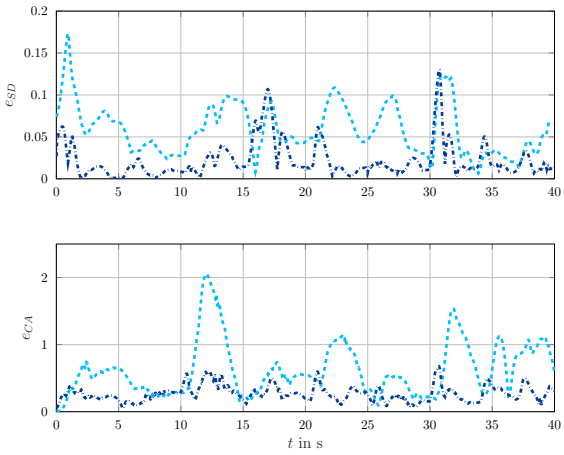

Fig. 7. Mean absolute error of swivel drives and continuum actuators. DOB-based controller: - · - . PID-like controller: - -.

$$e_{\mathrm{SD}} = \sum_{j=1}^{3} \frac{1}{3}|e_j|, \tag{19}$$

$$e_{\mathrm{CA}} = \sum_{j=4}^{7} \frac{1}{4}|e_j|, \tag{20}$$

where e_j is the j-th entry of $\boldsymbol{e}$. The results over time are presented in Figure 7. The average error over time of the swivel drive E_{SD} and continuum actuators E_{CA} is displayed in Table 4.2, where

$$E_{\mathrm{SD}} = \int_0^T \frac{1}{T} e_{\mathrm{SD}} \mathrm{d}t \tag{21}$$

$$E_{\mathrm{CA}} = \int_0^T \frac{1}{T} e_{\mathrm{CA}} \mathrm{d}t \tag{22}$$

and T is the duration of the trajectory.

	E_{SD}	E_{CA}
PID-like	0.0458	0.6388
DOB	0.0205	0.2716

A comparison of $\boldsymbol{\tau}_{\mathrm{ctrl}}$ of the continuum actuators for the DOB-based and the PID-like controller over time is shown in Figure 8.

5. DISCUSSION

In this section we discuss the experimental results first. Afterwards, we make comments on the implementation regarding computational time and tuning.

5.1 Discussion of the Experimental Results

The result presented in Figure 4 shows the overall behavior of the proposed control structure. We can see that, within 3 s, the controller is able to reach the desired state. Additionally, overshoots are within a reasonable range. The proposed controller converges to the desired value and can deal with external disturbances, which is backed up by the additional video material presented online (Müller (2019)). From the experimental results presented in Section 4.2 it is clear that the proposed DOB-based control structure shows superior performance in comparison to the PID-like controller presented in our previous work (Müller et al. (2020)).

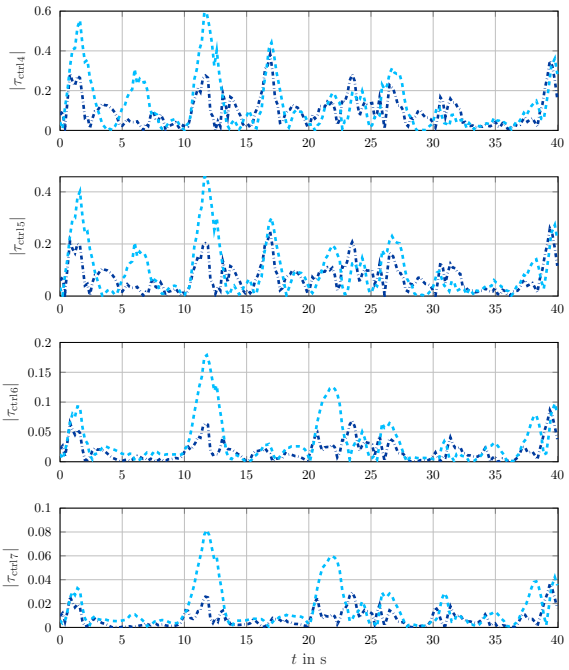

Fig. 8. Comparison of $\boldsymbol{\tau}_{\mathrm{ctrl}}$ for the continuum actuators for both concepts. Control action $\boldsymbol{\tau}_{\mathrm{ctrl}}$ of the DOB-based controller: - · - . Control action $\boldsymbol{\tau}_{\mathrm{ctrl}}$ of the PID-like controller: - -.

According to the Table 4.2, the performance of the rotational drives improves by 55% and the performance of the continuum actuators improves by 57%. Note that the parameters for the PD controller in our method are same as for the PID-like controller.

Taking a closer look at Figure 5, it can be observed that both the PID-like controller as well as the DOB-based controller have difficulties following the desired movements of the first joint q_1. However, the DOB-based controller shows a better performance for small movements, as it can be seen by q_1 from second 5 to 15. Additionally, during dynamical movements it can track its desired value better than the benchmark controller.

The same applies even stronger for the quasi continuum actuators presented in Figure 6, where we can see that the DOB-based controller reacts faster and with better precision compared to the PID-like controller. This is additionally emphasized by Figure 7, which clearly shows that the mean error of the DOB-based controller is almost always smaller. It comes as no surprise that the PD controller action $\boldsymbol{\tau}_{\mathrm{ctrl}}$ for our method is small compared to the benchmark PID-like controller as it is shown in Figure 8. The reason for this effect is the DOB. The behavior of the nominal system is forced onto the real plant by the DOB. Thus, a FF control for the nominal system is more likely to have the desired effect on the real system.

From Section 4.1 and 4.2 we conclude that the method can be applied on multiple QCM with different kinematics and a different sensor concept. Thus, we suppose that the presented approach is not only tailored for one particular QCM but it will generalize to a variety of CMs.

5.2 Computational Burden and Practical Implementation

For the Bionic Soft Arm with seven degrees of freedom the computational burden is quite high. The derivative $\frac{\partial}{\partial q}\boldsymbol{M}^{-1}(\boldsymbol{q})$ used in (10) requires the second derivative of the kinematics. Computing these terms online is expensive as it has been discussed in Falkenhahn et al. (2015). Furthermore, the proposed EKF has 21 states. Since, the complexity of the EKF grows exponentially with the dimension of the observer states, the extra 7 disturbance states $\hat{\boldsymbol{\xi}}$ drastically increase the computational effort. Although, this was not a problem for the Rapid Prototype DSpace used in our experiments, it might be a problem for weaker embedded controllers.
Besides the additional computational power required for the EKF, its difficulty of the practical implementation lies in the choice of the covariance matrices $\boldsymbol{Q}$ and $\boldsymbol{R}$. It is well known that EKFs are sensitive to their parameters. If we assume that $\boldsymbol{Q}$ and $\boldsymbol{R}$ are diagonal matrices, which is often done for EKF tuning, we are left with 28 tunable parameters. Since the ratio between $\boldsymbol{Q}$ and $\boldsymbol{R}$ determines the behavior of the EKF we need to determine 21 parameters. Tuning the proposed EKF was perceived as difficult due to the amount of parameters. A remedy in case of having trouble finding the right parameters could be Schneider and Georgakis (2013), since they proposes a systematic way of tuning the EKF as well as pointing out pitfalls.

6. CONCLUSION AND FUTURE WORK

A DOB-based control structure was proposed. We showed that the disturbance state $\boldsymbol{\xi}$ is observable. The DOB was realized using an EKF and the disturbance state was added to the control input. We implemented the control structure on two real robots, thus showing that our approach generalizes to multiple robots with different kinematics and sensors. The proposed controller structure outperformed a previous approach presented in Müller et al. (2020), which was based on a PID-like controller. We elaborated the significant improvements by the proposed structure and discussed the weaknesses of this method. These are in particular the computational burden and the difficulties in tuning the EKF since it is sensitive to its parameter choice.
There are multiple research directions to go from here. Obviously, a more efficient implementation and realization of the algorithm is desired. This might go together with a more straightforward tunable algorithm, which is either more robust to parameter variations or has fewer parameters. Another direction is to improve the proposed method itself. Yet, no assumptions have been made on the dynamics of the disturbance. We believe that an identified underlying behavior of the disturbance dynamic can improve the performance of the proposed controller even more.

REFERENCES

Albu-Schaffer, A., Eiberger, O., Grebenstein, M., Haddadin, S., Ott, C., Wimbock, T., Wolf, S., and Hirzinger, G. (2008). Soft robotics. *IEEE Robotics Automation Magazine*, 15(3), 20–30. doi:10.1109/MRA.2008.927979.

Chen, W., Yang, J., Guo, L., and Li, S. (2016). Disturbance-observer-based control and related methods an overview. *IEEE Transactions on Industrial Electronics*, 63(2), 1083–1095. doi: 10.1109/TIE.2015.2478397.

Chen, W.H., Ballance, D.J., Gawthrop, P.J., and O'Reilly, J. (2000). A nonlinear disturbance observer for robotic manipulators. *IEEE Transactions on Industrial Electronics*, 47(4), 932–938. doi:10.1109/41.857974.

Falkenhahn, V., Mahl, T., Hildebrandt, A., Neumann, R., and Sawodny, O. (2014). Dynamic modeling of constant curvature continuum robots using the euler-lagrange formalism. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2428–2433. doi:10.1109/IROS.2014.6942892.

Falkenhahn, V., Mahl, T., Hildebrandt, A., Neumann, R., and Sawodny, O. (2015). Dynamic modeling of bellows-actuated continuum robots using the euler-lagrange formalism. *IEEE Transactions on Robotics*, 31(6), 1483–1496. doi:10.1109/TRO.2015.2496826.

Laschi, C., Mazzolai, B., and Cianchetti, M. (2016). Soft robotics: Technologies and systems pushing the boundaries of robot abilities. *Sci. Robot*, 1(1), eaah3690.

Luenberger, D.G. (1964). Observing the state of a linear system. *IEEE Transactions on Military Electronics*, 8(2), 74–80. doi:10.1109/TME.1964.4323124.

Müller, D., Raisch, A., Hildebrandt, A., and Sawodny, O. (2020). Nonlinear model based dynamic control of pneumatic driven quasicontinuum manipulators. In *2020 IEEE/SICE International Symposium on System Integration*.

Müller, D. (2019). Disturbance observer based control for quasi continuum manipulators. URL `https://www.youtube.com/playlist?list=PLcK7G5QrmRsZ_qspF6mi3Jym7Bkpk66qB`.

Müller, D. (2020). Disturbance observer based control for quasi continuum manipulators. URL `https://github.com/Daniel-Mller/DOBC_for_QCM-.git`.

Ohishi, K. (1983). Torque-speed regulation of dc motor based on load torque estimation. *IEEJ International Power Electronics Conference, IPEC-TOKYO, 1983-3*, 2, 1209–1216.

Polygerinos, P., Correll, N., Morin, S.A., Mosadegh, B., Onal, C.D., Petersen, K., Cianchetti, M., Tolley, M.T., and Shepherd, R.F. (2017). Soft robotics: Review of fluid-driven intrinsically soft devices; manufacturing, sensing, control, and applications in human-robot interaction. *Advanced Engineering Materials*, 19(12), 1700016.

Schneider, R. and Georgakis, C. (2013). How to not make the extended kalman filter fail. *Industrial & Engineering Chemistry Research*, 52(9), 3354–3362.

Terejanu, G.A. et al. (2008). Extended kalman filter tutorial. *University at Buffalo*.

Thuruthel, T.G., Ansari, Y., Falotico, E., and Laschi, C. (2018). Control strategies for soft robotic manipulators: A survey. *Soft Robotics*, 5(2), 149–163. doi: 10.1089/soro.2017.0007. PMID: 29297756.

Webster, R.J. and Jones, B.A. (2010). Design and kinematic modeling of constant curvature continuum robots: A review. *The International Journal of Robotics Research*, 29(13), 1661–1683. doi: 10.1177/0278364910368147.