Stiffness Control of Deformable Robots Using Finite Element Modeling

Margaret Koehler, Allison M Okamura, Christian Duriez

Stiffness Control of Deformable Robots Using Finite Element Modeling

Margaret Koehler[1], Allison M. Okamura[1], and Christian Duriez[2]

***Abstract*—Due to the complexity of modeling deformable materials and infinite degrees of freedom, the rich background of rigid robot control has not been transferred to soft robots. Thus, most model-based control techniques developed for soft robots and soft haptic interfaces are specific to the particular device. In this paper, we develop a general method for stiffness control of soft robots suitable for arbitrary robot geometry and many types of actuation. Extending previous work that uses finite element modeling for position control, we determine the relationship between end-effector and actuator compliance, including the inherent device compliance, and use this to determine the appropriate controlled actuator stiffness for a desired stiffness of the end-effector. Such stiffness control, as the first component of impedance control, can be used to compensate for the natural stiffness of the deformable device and to control the robot's interaction with the environment or a user. We validate the stiffness projection on a deformable robot and include this stiffness projection in a haptic control loop to render a virtual fixture.**

***Index Terms*—Modeling, Control, and Learning for Soft Robots; Compliance and Impedance Control; Haptics and Haptic Interfaces**

I. Introduction

BY their natural compliance, some robots create motion by deformation, rather than the sole use of translation or rotation of joints. We refer to these as *deformable robots*, following the naming of deformable solids in mechanics. These robots include both devices made of soft materials like silicone [1] and those which are deformable due to their structure. For example, concentric tube robots [2] can be made of metal alloy, but their thin rod structure makes deformation an important element of their actuation. Such deformable robots are particularly appropriate for contact with the environment, since their inherit compliance can make them safer and more robust than rigid robots. The stiffness of the robot end-effector is dependent on the material properties, the robot structure, the controlled stiffness of the actuation, and other constraints. Controlling this stiffness has the potential to make these robots even more versatile, but also presents significant challenges.

One of the advantages of using deformation instead of rigid joints for the transmission of motion is that friction and backlash can be avoided. These nonlinear effects can significantly impact the performance of a robot arm or the transparency of a haptic device. However, with deformation, the behavior of the material cannot be ignored, as well as all the factors that will influence the deformation, in particular the geometry and the boundary conditions. Some numerical methods, like finite element modeling (FEM), make it possible to account for the material's behavior during deformations at the scale of the robot structure and can be adapted to most geometries and boundary conditions.

A. Contributions

In this paper, we develop a general formulation of stiffness control for deformable robots made of elastic material. The formulation uses finite elements to analyze the natural stiffness of the robot structure and the stiffness transmission between effectors and actuators. Three contributions are presented:

- A methodology to project stiffness from end-effectors to actuators through a compliant device using general FEM and beam models.
- A methodology to estimate the end-effector force and position using only actuator force and position on a backdrivable deformable robot.
- Demonstration of a novel deformable robot/haptic device that benefits from the new control methods described above.

B. Related Work

Much analysis has been done on stiffness mapping between actuator and end-effector coordinates for robots with rigid links. This takes two main forms: active impedance control [3] using a fast feedback loop and passive (but sometimes controllable) compliance in the joints. The dependence of Cartesian end-effector stiffness on robot configuration has been studied in detail including for redundant [4], [5], [6] and parallel manipulators [7]. Firouzeh et al. used adjustable stiffness joints to control the compliance of a gripper [8]. Albu-Schaffer et al. focused on passively compliant joints in [9]. Active and passive impedance control have also been explored in [10], [11], [12]. For haptics, Gillespie et al. analyzed flexure joints for a haptic pantograph and determined the actuation required to compensate for the return-to-center behavior of

Manuscript received: September 10, 2018; Accepted: December 6, 2018.

This paper was recommended for publication by Editor Kyu-Jin Cho upon evaluation of the Associate Editor and Reviewers' comments. This work was supported by a National Science Foundation Graduate Fellowship, a Chateaubriand Fellowship, a Fulbright Visiting Scholar grant, National Science Foundation grant 1830163, and FEDER (European Regional Development Fund) through the region Haut-de-France, grant COMOROS.

[1]M. Koehler and A.M. Okamura are with the Department of Mechanical Engineering, Stanford University, Stanford, CA 94305 USA mkoehler@stanford.edu, aokamura@stanford.edu

[2]C. Duriez is with INRIA, Defrost Team, and the University of Lille, France christian.duriez@inria.fr

Digital Object Identifier (DOI): see top of this page.

the joints [13]. Herzig et al. used a stiffness mapping for a pneumatically controlled rigid link haptic device [14].

Less work has been done thus far in the area of deformable robot stiffness control. This is in part because deformable robots have an inherent compliance that passively moderates interactions with the environment. Mahvash et al. demonstrated a stiffness controller for a concentric tube continuum manipulator based on the Cosserat rod model [15]. In that work, the end-effector position was sensed directly, and the tip force was controlled to produce a specific apparent stiffness. Della Santina et al. [16] present an impedance controller for a soft continuum robot using a Piecewise Constant Curvature assumption to link the soft robot to an equivalent rigid robot representation. In the soft robot community, numerous works have focused on obtaining actuators with variable stiffness using, for example, particle jamming [17], shape memory polymers [18], magnetorheology [19], and antagonistic soft actuation [20]. These approaches allow local modification and control of stiffness, but do not necessarily take into account the compliance of the whole structure of the robot or consider the apparent end-effector stiffness. Largilliere et al. [21] used a finite element model to compute the mechanics of the robot, but the force modulation at the end-effector requires direct position sensing, and no projection between actuator and end-effector stiffness was considered.

II. Stiffness Control Thoery

In this section, we derive the projection from end-effector stiffness to actuator stiffness. First, we summarize the constraint formulation described in [22] which allows the derivation in real-time of the inner compliance of the deformable robot using computational mechanics. From this basis, we first determine the apparent end-effector stiffness given a known actuator stiffness, i.e. we solve a forward problem. Then, we invert the result to obtain the appropriate actuator stiffness for a given target end-effector stiffness.

A. Background on Reduced Compliance Formulation

First, we summarize the derivation of the reduced compliance matrix and explain how it is used for soft robot control. The finite element method is used to model the mechanical behavior of the robot from its material and geometry. The internal forces on the robot are modeled over time through the following equation:

$$\mathbf{f}(\mathbf{x}_i) = \mathbf{f}(\mathbf{x}_{i-1}) + \mathbf{K}(\mathbf{x}_{i-1})\Delta\mathbf{x} \tag{1}$$

in which i is the time step, $\mathbf{f}$ is the vector of internal forces, $\mathbf{x}$ is the state vector consisting of the degrees of freedom of the nodes, $\Delta\mathbf{x} = \mathbf{x}_i - \mathbf{x}_{i-1}$, and $\mathbf{K}(\mathbf{x}_{i-1})$ is the configuration-dependent stiffness matrix evaluated at a given state from the finite elements. In this work, we consider a quasi-static model, so no mass or damping is considered. In Section IV, we identify the limits brought by this hypothesis and propose possible ways to address this in the future.

By imposing the assumption of static equilibrium, we require that the internal forces ($\mathbf{f}(\mathbf{x}_i)$) and external forces ($\mathbf{f}_e$) balance.

$$\mathbf{f}(\mathbf{x}_i) + \mathbf{f}_e = 0 \tag{2}$$

External forces on the robot consist of known forces and constraints that provide a relationship between the force on the robot and a displacement at the node where the force is applied. For example, a rigid constraint will supply whatever force is necessary so that a particular node stays in the same position. In the case of deformable robots, actuators and end-effectors are considered constraints. We use a mapping function, $\boldsymbol{\delta}(\mathbf{x})$, to translate node positions into meaningful constraint quantities, such as end-effector position relative to a target position. This mapping function can account for constraints (including end-effectors and actuators) that act on more than one node or which rotate relative to the node frame during a motion and is in general non-linear with respect to the nodes. Additionally, we define $\mathbf{H} = \frac{\partial\boldsymbol{\delta}}{\partial\mathbf{x}}$, which captures the direction that the constraint can apply loads. In the case shown later in this paper, the actuator and end-effector directions are coincident with single nodes and do not rotate relative to the nodes, so $\boldsymbol{\delta}(\mathbf{x})$ is linear in $\mathbf{x}$, and $\mathbf{H}$ is constant.

By substituting the linearized FEM for $\mathbf{f}(\mathbf{x}_i)$ and splitting the external forces between constraint forces and known external forces (such as gravity), we obtain the following equation:

$$\mathbf{f}(\mathbf{x}_{i-1}) + \mathbf{K}(\mathbf{x}_{i-1})\Delta\mathbf{x} + \mathbf{p} - \mathbf{H}^T\boldsymbol{\lambda} = 0 \tag{3}$$

where $\mathbf{p}$ includes all known external forces and $\mathbf{H}^T\boldsymbol{\lambda}$ gathers the contribution of the constraints with $\boldsymbol{\lambda}$ representing the amount of effort applied by the constraint.

The static equilibrium equation can then be projected into the constraint space in order to reduce the dimension of the equation that must be solved, from that of the full robot state to that of only the constraints. First, let $\Delta\mathbf{x} = \Delta\mathbf{x}^{\text{free}} + \Delta\mathbf{x}^\star$ such that

$$\begin{aligned} \mathbf{f}(\mathbf{x}_{i-1}) + \mathbf{K}(\mathbf{x}_{i-1})\Delta\mathbf{x}^{\text{free}} + \mathbf{p} &= 0 \\ \mathbf{K}(\mathbf{x}_{i-1})\Delta\mathbf{x}^\star - \mathbf{H}^T\boldsymbol{\lambda} &= 0 \end{aligned} \tag{4}$$

Then, the mapping $\boldsymbol{\delta}(\mathbf{x})$ can be written as

$$\begin{aligned} \boldsymbol{\delta}(\mathbf{x}_i) &= \boldsymbol{\delta}(\mathbf{x}_{i-1}) + \mathbf{H}(\Delta\mathbf{x}^{\text{free}} + \Delta\mathbf{x}^\star) \\ &= \boldsymbol{\delta}^{\text{free}} + \mathbf{H}\Delta\mathbf{x}^\star \\ &= \boldsymbol{\delta}^{\text{free}} + \mathbf{H}\mathbf{K}^{-1}\mathbf{H}^T\boldsymbol{\lambda} \end{aligned} \tag{5}$$

with $\boldsymbol{\delta}^{\text{free}} = \boldsymbol{\delta}(\mathbf{x}_{i-1}) + \mathbf{H}\Delta\mathbf{x}^{\text{free}}$.

We substitute $\mathbf{W} = \mathbf{H}\mathbf{K}^{-1}\mathbf{H}^T$. Thus the relevant mechanics of the robot are captured in the following equation:

$$\boldsymbol{\delta} = \mathbf{W}\boldsymbol{\lambda} + \boldsymbol{\delta}^{\text{free}} \tag{6}$$

To obtain an efficient computation of $\mathbf{W}$, $\mathbf{K}^{-1}$ is not explicitly computed. Rather, $\mathbf{K}$ is first factorized, then $\mathbf{K}^{-1}\mathbf{H}^T$ is computed and stored in a matrix which is multiplied by $\mathbf{H}$.

B. Stiffness Projection

To perform the projection from a desired end-effector stiffness to a controlled actuator stiffness, we are interested in the reduced compliance equation for the actuator and end-effector constraints (denoted $(*)_a$ and $(*)_e$, respectively).

$$\begin{bmatrix}\boldsymbol{\delta}_e \\ \boldsymbol{\delta}_a\end{bmatrix} = \begin{bmatrix}\mathbf{W}_{ee} & \mathbf{W}_{ea} \\ \mathbf{W}_{ae} & \mathbf{W}_{aa}\end{bmatrix}\begin{bmatrix}\boldsymbol{\lambda}_e \\ \boldsymbol{\lambda}_a\end{bmatrix} + \begin{bmatrix}\boldsymbol{\delta}_e^{free} \\ \boldsymbol{\delta}_a^{free}\end{bmatrix} \tag{7}$$

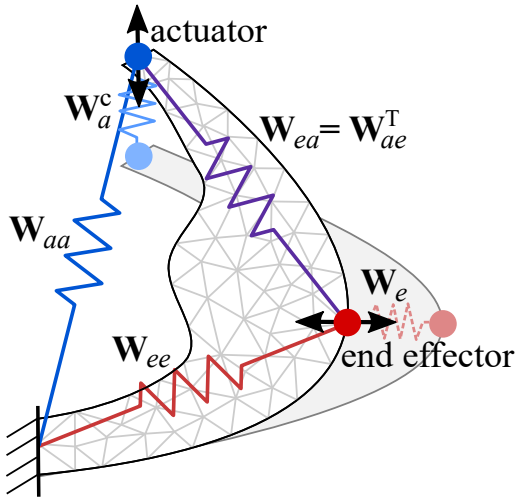


Fig. 1: A generic one-degree-of-freedom deformable robot is moved from one configuration to another. The apparent compliance of the end-effector, $\mathbf{W}_e$, depends on the controlled compliance of the actuator, $\mathbf{W}_a^c$, as well as the inherent compliance of the device, described by the components of the reduced compliance matrix, $\mathbf{W}_{ee}$, $\mathbf{W}_{ea}$, $\mathbf{W}_{ae}$, and $\mathbf{W}_{aa}$.

and

$$\begin{bmatrix} \mathbf{W}_{ee} & \mathbf{W}_{ea} \\ \mathbf{W}_{ae} & \mathbf{W}_{aa} \end{bmatrix} = \begin{bmatrix} \mathbf{H}_e\mathbf{K}^{-1}\mathbf{H}_e^T & \mathbf{H}_e\mathbf{K}^{-1}\mathbf{H}_a^T \\ \mathbf{H}_a\mathbf{K}^{-1}\mathbf{H}_e^T & \mathbf{H}_a\mathbf{K}^{-1}\mathbf{H}_a^T \end{bmatrix} \tag{8}$$

The terms of the compliance matrix can be interpreted as follows. $\mathbf{W}_{ee}$ is the compliance matrix of the end-effector degrees of freedom relative to the fixed world coordinate system, taking into account the fixed points of the deformable body. The off-diagonal terms include any coupling between degrees of freedom of the end-effector. Similarly, $\mathbf{W}_{aa}$ is the compliance matrix of the actuator degrees of freedom relative to the fixed world coordinate system. This is the matrix that we will modify by controlling the stiffness of the actuators. $\mathbf{W}_{ea} = \mathbf{W}_{ae}^T$ capture the coupling in compliance between the actuators and end-effectors. These matrices are important for evaluating force transmission between the actuators and end-effectors. Without sufficient coupling, the end-effector will not be controllable in all directions. This singularity is described in [23]. Figure 1 shows these compliances for a generic one-degree-of-freedom deformable robot.

For position control of the end-effector, as presented in [24], the following quadratic program (QP) can be solved

$$\min_{\boldsymbol{\lambda}_a} \quad \frac{1}{2}\boldsymbol{\lambda}_a^T\mathbf{W}_{ea}^T\mathbf{W}_{ea}\boldsymbol{\lambda}_a + \boldsymbol{\lambda}_a^T\mathbf{W}_{ea}^T\boldsymbol{\delta}_e^{\text{free}} \tag{9}$$

In this case, $\boldsymbol{\delta}_e$ is defined as the position of the end-effector relative to a target position. For actuators we can consider constraints on the maximum displacement ($\boldsymbol{\delta}_a$) and maximum force ($\boldsymbol{\lambda}_a$) in either direction. Thus, for a uni-directional transmission like a cable, we could limit the force to only allow tension and not compression forces. For the end-effector, we can drive either the force or displacement to a desired value.

A given configuration consisting of actuator and end-effector forces and positions (denoted $(*)^o$) will satisfy the linearized static equilibrium of Equation 7. This configuration and the forces and positions related to it will be referred to as the *bias* configuration. We can then look at small deviations in position and force relative to this solution (denoted $\Delta(*)$) to evaluate the stiffness of the end-effector. These deviations are coupled by the reduced compliance matrix.

$$\begin{bmatrix} \boldsymbol{\delta}_e^o + \Delta\boldsymbol{\delta}_e \\ \boldsymbol{\delta}_a^o + \Delta\boldsymbol{\delta}_a \end{bmatrix} = \begin{bmatrix} \mathbf{W}_{ee} & \mathbf{W}_{ea} \\ \mathbf{W}_{ae} & \mathbf{W}_{aa} \end{bmatrix} \begin{bmatrix} \boldsymbol{\lambda}_e^o + \Delta\boldsymbol{\lambda}_e \\ \boldsymbol{\lambda}_a^o + \Delta\boldsymbol{\lambda}_a \end{bmatrix} + \begin{bmatrix} \boldsymbol{\delta}_e^{free} \\ \boldsymbol{\delta}_a^{free} \end{bmatrix} \tag{10}$$

$$\begin{bmatrix} \Delta\boldsymbol{\delta}_e \\ \Delta\boldsymbol{\delta}_a \end{bmatrix} = \begin{bmatrix} \mathbf{W}_{ee} & \mathbf{W}_{ea} \\ \mathbf{W}_{ae} & \mathbf{W}_{aa} \end{bmatrix} \begin{bmatrix} \Delta\boldsymbol{\lambda}_e \\ \Delta\boldsymbol{\lambda}_a \end{bmatrix} \tag{11}$$

This formulation does not take into account variation of $\mathbf{H}$ or $\mathbf{W}$, since it is based on $\mathbf{K}$ which is the linearized stiffness from the finite element model. To account for non-linear stiffness in the device, the reduced compliance can be updated as the device configuration changes. The actuators are controlled to render a stiffness, $\mathbf{K}_a^c$. For a typical negative stiffness, the force exerted by the actuator on the deformable body is in the opposite direction of the change of position. Similarly, the actuator compliance is $\mathbf{W}_a^c = (\mathbf{K}_a^c)^{-1}$.

$$\mathbf{K}_a^c\Delta\boldsymbol{\delta}_a = \Delta\boldsymbol{\lambda}_a \tag{12}$$

Substituting for $\Delta\boldsymbol{\delta}_a$ in Equation 11 and solving for $\Delta\boldsymbol{\delta}_e$, we obtain the apparent compliance of the end-effector, $\mathbf{W}_e$.

$$\begin{aligned} \Delta\boldsymbol{\delta}_a &= \mathbf{W}_a^c\Delta\boldsymbol{\lambda}_a = \mathbf{W}_{ae}\Delta\boldsymbol{\lambda}_e + \mathbf{W}_{aa}\Delta\boldsymbol{\lambda}_a \\ \Delta\boldsymbol{\delta}_e &= (\mathbf{W}_{ee} + \mathbf{W}_{ea}(\mathbf{W}_a^c - \mathbf{W}_{aa})^{-1}\mathbf{W}_{ae})\Delta\boldsymbol{\lambda}_e \\ \mathbf{W}_e &= \mathbf{W}_{ee} + \mathbf{W}_{ea}(\mathbf{W}_a^c - \mathbf{W}_{aa})^{-1}\mathbf{W}_{ae} \end{aligned} \tag{13}$$

C. Special cases and Inversion

A few special cases of $\mathbf{W}_a^c$ are of particular interest.

- Case 1: In pure force control, we have $\mathbf{W}_a^c \to \infty$ that leads to $\mathbf{W}_e \to \mathbf{W}_{ee}$. As explained previously, this term provides the compliance of the deformable robot body at the end-effector. It takes into account the bindings of the robot and is valid around the current configuration.
- Case 2: For perfectly stiff position actuators where $\mathbf{W}_a^c = 0$, we obtain $\mathbf{W}_e = \mathbf{W}_{ee} - \mathbf{W}_{ea}\mathbf{W}_{aa}^{-1}\mathbf{W}_{ae}$. Thus, this $\mathbf{W}_e^{-1}$ is the maximum stiffness we can obtain at the end-effector in this robot configuration.
- Case 3: We can consider positive stiffness actuation, in which the actuator applies force to the body in the same direction as the displacement that is sensed. While unstable on its own, such a stiffness is not unstable when coupled to the deformable body as long as $\mathbf{W}_{ee} + \mathbf{W}_{ea}(\mathbf{W}_a^c - \mathbf{W}_{aa})^{-1}\mathbf{W}_{ae} > 0$. Such a stiffness could increase the compliance of the end-effector beyond its inherent compliance, $\mathbf{W}_{ee}$, and be useful for canceling the inherent stiffness characteristics of the device. However, care must be taken when canceling the stiffness since we use a linearization of the true stiffness.

By inverting Equation 13, we can determine the necessary actuator stiffness, $\mathbf{K}_a^c$, to achieve a desired end-effector compliance, $\mathbf{W}_e^{\text{d}} = (\mathbf{K}_e^{\text{d}})^{-1}$. For simplicity, we limit our analysis to systems with the same number of actuator and end-effector degrees of freedom.

The necessary controlled actuator stiffness is computed by

$$\mathbf{K}_a^c = (\mathbf{W}_{aa} + \mathbf{W}_{ae}(\mathbf{W}_e^{\mathrm{d}} - \mathbf{W}_{ee})^{-1}\mathbf{W}_{ea})^{-1} \tag{14}$$

This stiffness is limited in a range of feasible compliances, in particular $\mathbf{W}_e^{\mathrm{d}}$ is limited by the maximum stiffness described above in Case 2. We assume that we are in a non-singular configuration [23], and that the controlled actuator stiffness can be coupled across actuators, allowing $\mathbf{K}_a^c$ to be a full matrix with terms on the off-diagonals. For example, if the actuators are a set of DC motors run as torque sources in a fast control loop with feedback on their position, the displacement of one motor could lead to a torque response on another motor. If instead, the stiffness of the actuators cannot be coupled, $\mathbf{K}_a^c$ is limited to a diagonal matrix and many more actuators are required to fully control $\mathbf{W}_e$.

III. Experimental Evaluation

There are many potential use cases of the stiffness control method described above, given its genericity. In this section, we present a first experimental evaluation on a backdrivable two-degree-of-freedom deformable robot/haptic device. The robot is controlled to demonstrate the stiffness projection, and then that stiffness projection is used in the robot control loop.

A. Design and Modeling

The robot, shown in Figure 2, consists of a flexible curved beam driven by two motors (Maxon RE25) via capstan drives. The motors rotate the two ends of the beam, creating force and deformation at the end-effector. The flexible beam is made of polycaprolactone (PCL) printed on a relatively inexpensive 3D printer (Makerbot Replicator 2). The cross-section of the beam is 2 mm $\times$ 10 mm. The motors are current-controlled from an Arduino Uno through Advanced Motion Controls PWM Servo Amplifiers. Actuator angle is sensed via 2000-count quadrature encoders on the motors. Consisting of a single 3D printed part, the deformable linkage is easy to assemble and alternate designs could be made easily, which underscores the general applicability of the FEM approach.

The motors are run as open-loop torque sources, and the Arduino controls them at a high rate (approximately 1.2 kHz) to render a stiffness based on the measured motor position. The stiffness of the motors can be coupled through the off-diagonal terms of the stiffness matrix.

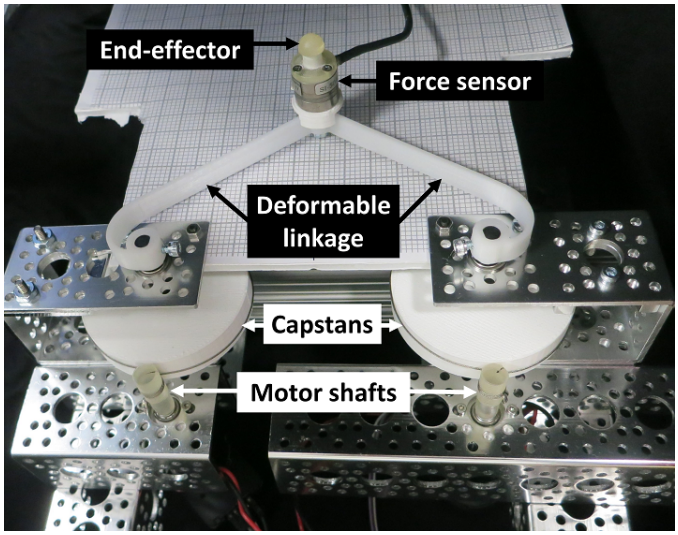
The device is modeled and simulated using the SOFA Framework [24]. The model is based on beam elements and consists of 21 nodes with 6 degrees of freedom per node [25]. The implementation accounts for large displacements.

B. State Estimation for Backdrivable Deformable Robots

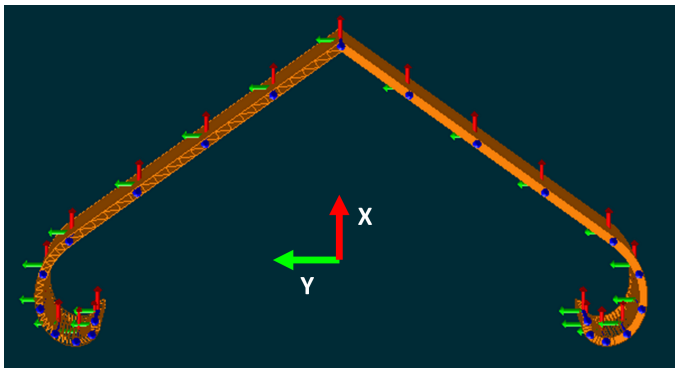
Similar to impedance-type control schemes typically used with haptic devices, the control method described below depends on the position of the end-effector. Like rigid robots, if the deformable robot is backdrivable, we can, in theory, find the position of the end-effector given the measures made on the actuator space. However, unlike a robot (or haptic device) with rigid links and free joints, there is not a simple kinematic relationship between the actuator and end-effector positions. Instead, we must take into account the deformation of the robot due to both actuator and end-effector forces. To estimate the positions (and forces) of the end-effector, then, the commanded motor torque, the motor position, and the deformable mechanical model are used, rather than relying only on the actuator positions (as is done for rigid robots). This estimator is based on an optimization that consists of minimizing the gap between the measured $\boldsymbol{\delta}_a^m$ and simulated actuator positions $\boldsymbol{\delta}_a$. Assuming the model is predictive, we optimize the only mechanical unknown, which is the force applied by the user on the end-effector, $\boldsymbol{\lambda}_e$.

We use a quadratic programming approach similar to that described in [24] and expressed in Equation 15.

$$\min_{\boldsymbol{\lambda}_e}(\boldsymbol{\delta}_a - \boldsymbol{\delta}_a^m)^2 = \min_{\boldsymbol{\lambda}_e} \quad \frac{1}{2}\boldsymbol{\lambda}_e^T\mathbf{W}_{ae}^T\mathbf{W}_{ae}\boldsymbol{\lambda}_e + \boldsymbol{\lambda}_e^T\mathbf{W}_{ae}^T\boldsymbol{\delta}_a^{\mathrm{free}} \tag{15}$$



(a)



(b)

Fig. 2: The deformable robot (a) consists of a flexible linkage driven by two motors via capstan drives. The force sensor attachment was used to measure the forces applied to the end-effector. (b) The beam element model is shown. The degrees of freedom, consisting of the end nodes of the beams, are marked with coordinate axes.

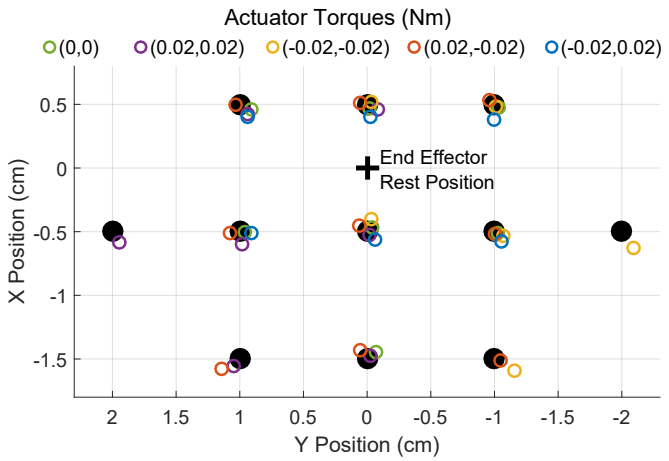


Fig. 3: Position sensing results for the estimator. The actual positions are plotted in black while the estimated positions are plotted in different colors based on applied actuation torques. The rest position with no actuator torques is set as the origin. The rest position (no end-effector forces) varies with applied actuation torque. The estimator was tested at points near these rest positions, hence the different sets of target locations for different actuation torques.

The motor torques are applied as constant torques in the model and thus included in $\boldsymbol{\delta}_a^{\text{free}}$. In our implementation, the optimization is unconstrained, and the minimum is unique since the number of actuator and end-effector degrees of freedom are equal. The formula for $\boldsymbol{\delta}_a$ is based on the reduced compliance formulation described in Section II-A. The estimated end-effector force is used to compute $\boldsymbol{\delta}_e$ and determine the position of the end-effector.

The estimation algorithm was validated experimentally by moving the end-effector to known positions on a physical grid and comparing the estimated end-effector position under five different actuator loadings (i.e., combinations of applied motor torques). The Young's modulus of the model (450 MPa) was calibrated so that when torques were applied to the motor with no force at the end-effector, the estimated force was close to zero for five different loading configurations. The results of the estimation experiment are shown in Figure 3. The average error was 0.78 mm over a workspace of 30 mm $\times$ 20 mm.

C. Stiffness Projection Validation

To validate the stiffness projection formula derived in Section II-B, the stiffness of the device was measured under a variety of actuation stiffnesses. The motors were commanded to render particular stiffnesses via a fast feedback control loop according to the following equation

$$\boldsymbol{\lambda}_a = \mathbf{K}_a^c(\boldsymbol{\delta}_a - \boldsymbol{\delta}_a^o) + \boldsymbol{\lambda}_a^o \tag{16}$$

while the end-effector was moved in the two Cartesian end-effector directions, X and Y. Recall that $\boldsymbol{\delta}_a^o$ and $\boldsymbol{\lambda}_a^o$ refer to the bias position and torque of the actuator, which define the bias configuration around which the stiffness linearization was made. The position trace of the end-effector was taken from positions estimated by the model, and the maximum

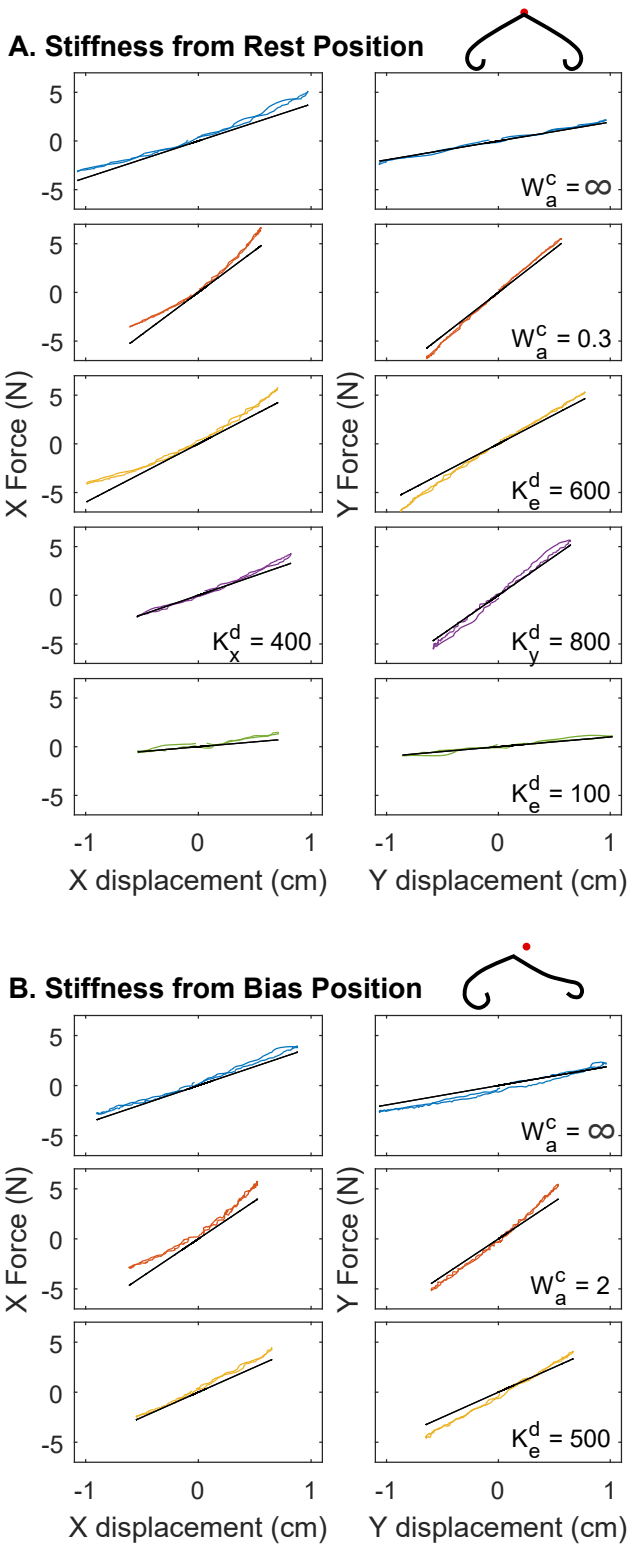


Fig. 4: The projected stiffness formulation was experimentally verified in the rest configuration (A) and a bias configuration (B). Perfectly compliant actuation, highly stiff actuation, and specified end-effector stiffness, including isometric, non-isometric, and stiffness cancellation cases, are shown. Unless otherwise noted, either actuator compliance or end-effector stiffness were isometric and diagonal. The actuator stiffness predicted by Eq. 13 is plotted in black. Stiffness units are N/m and compliance units are rad/Nm.

displacement in each direction was confirmed on the real device by measuring movement with respect to a physical grid placed beneath the end-effector. The force was measured using an ATI Nano 17 force/torque sensor (Industrial Automation).

The results are shown in Figure 4 for two positions: the natural rest position, in which the bias force and position of the actuators are 0, and a different configuration in which the bias force and position were computed for a given end-effector position (x = -4 mm, y = 6 mm). The stiffness was tested under the following conditions: no actuator stiffness (Case 1), high actuator stiffness (Case 2), controlled stiffness (Case 3, and Equation 14) for an isometric stiffness, for non-isometric stiffness, and for stiffness cancellation. For stiffness from the bias position, the bias actuator force was first applied to the device with no stiffness. Then the stiffness control was started. Due to minor discrepancies between the model and the actual device, the initial end-effector position changed upon addition of the stiffness control. Displacement was measured from this new initial position. These results confirm the stiffness projection derived in Section II-B. The results show that the stiffness is non-linear, especially in the x direction, but the tangent stiffness matches the target stiffness well in all cases.

D. Haptic Virtual Fixture Demonstration

Finally, the stiffness control is demonstrated in a haptic virtual fixture task. As described in the introduction, deformable robots provide the advantage of avoiding joints which can add friction and backlash to the system. This is particularly important for haptic devices. However, to render high stiffness stably, haptic devices require fast feedback control. The complex modeling required for deformable devices make such high rates of control difficult. Thus, we propose using the stiffness control as a local linearization of the haptic response that can be rendered in between slower control steps that account for the non-linear deformation response of the robot. The complete control loop is shown in Figure 5. In some ways, the force cancellation step is a combination of the typical impedance type and admittance type haptic control approaches for rigid robots, since forces and deformations are inherently coupled in a deformable device.

The virtual fixture was intended to encourage the user to stay on the y axis while making movement along that axis as free as possible. Two haptic conditions were tested and compared to the natural device dynamics. Both haptic conditions consisted of the same basic force cancellation, and the second one added a local stiffness control component. The purpose of the force cancellation was to set the actuator torques such that no force would be felt by the user while they were on the y axis. As the user moves away from this axis, the natural or controlled stiffness (depending on the haptic condition) of the end-effector resists their motion. The force cancellation was computed as follows, similar to that in [21].

We use the state estimation to get the actual position of the end-effector based on the measured position of the actuators. The estimated position of the end-effector was projected onto the y axis and filtered (Equation 19, explained below) to obtain the bias end-effector position for the controller. The bias end-effector force is set to 0. The controller simulation then solves

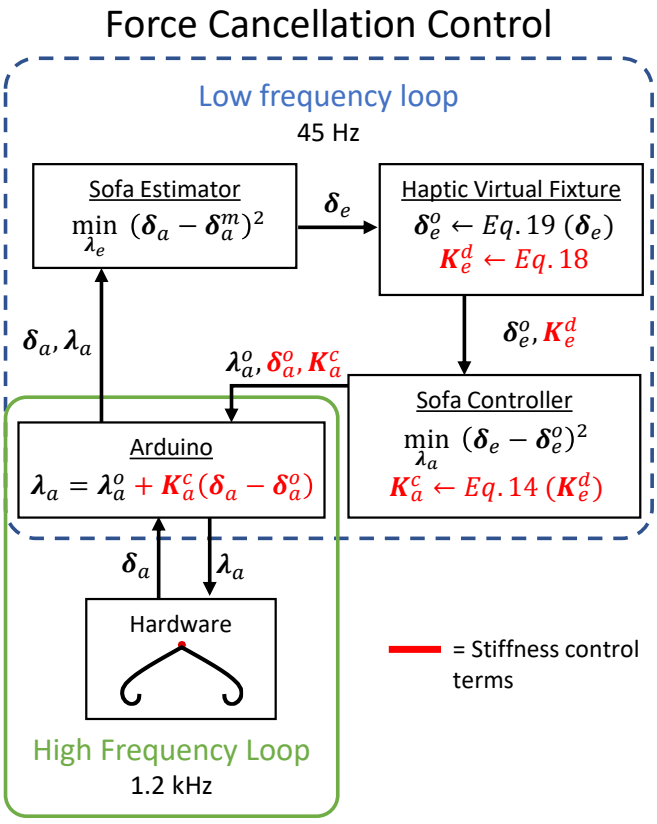


Fig. 5: The haptic control loop consists of a high frequency loop which controls the motors to render the commanded stiffness and a low frequency loop which updates the linearized stiffness model based on the beam model. The slower loop estimates the end-effector position, uses this position to compute the force and stiffness that should be displayed, and determines the motor bias torques, positions, and stiffnesses to command. Components of the control which are only used in the stiffness control case are shown in red.

the QP in Equation 17, which minimizes the distance error between the end-effector and the bias position.

$$\min_{\boldsymbol{\lambda}_a}(\boldsymbol{\delta}_e - \boldsymbol{\delta}_e^o)^2 = \min_{\boldsymbol{\lambda}_a} \quad \frac{1}{2}\boldsymbol{\lambda}_a^T \mathbf{W}_{ea}^T \mathbf{W}_{ea} \boldsymbol{\lambda}_a + \boldsymbol{\lambda}_a^T \mathbf{W}_{ea}^T \boldsymbol{\delta}_e^{\text{free}} \tag{17}$$

The solution of the QP provides the bias actuation torques and positions.

In the force cancellation without stiffness case, only the bias torques were used, and the natural stiffness of the device remains. In the force cancellation with stiffness case, an additional stiffness command was used. Using Equation 14, the controller simulation determined the required actuator stiffness to render the following end-effector stiffness matrix, with high stiffness in the x direction and low stiffness (lower than the natural device stiffness) in the y direction:

$$\mathbf{K}_e^d = \begin{bmatrix} 700 & 0 \\ 0 & 100 \end{bmatrix} \text{N/m} \tag{18}$$

This stiffness should make the motion more free in the y direction and the virtual fixture stiffer in the x direction. The

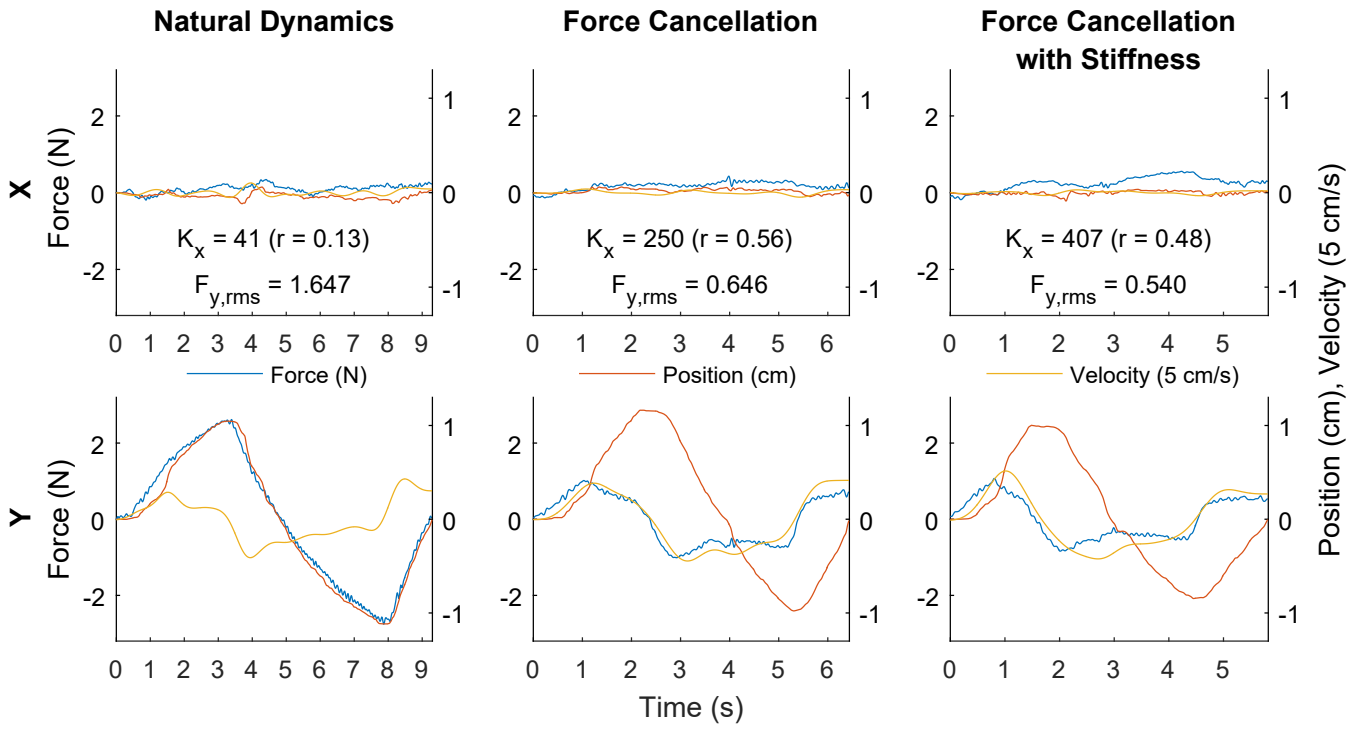


Fig. 6: The control loop was tested in two cases (force cancellation with and without stiffness control) and compared to the natural dynamics. The position of the end-effector was measured using the estimator, and the velocity was computed from the position and then filtered. The end-effector forces were measured. To compare the cancellation of force in the y direction, the root mean square was computed. The stiffness in the x direction, which should be high to keep the user on the y axis, is approximated using a linear fit between x force and position. The force cancellation cases reduced the force in the y direction, canceling the natural stiffness of the device. However, a damping force remains, due to the filtering required to stabilize the controller. The stiffness in the x direction was higher, as desired, when stiffness control was added to the force cancellation.

Arduino then implements the fast control loop to render the computed stiffness, $\mathbf{K}_a^c$, around the bias actuator position and force as in Equation 16.

As stated, the bias end-effector position used in the controller was first filtered. This was necessary to make the control loop stable, since this loop is running at a low rate relative to the stiffness being rendered. Using an exponentially weighted moving average with $\alpha = 0.5$, the filtering and projection onto the y axis are then:

$$\begin{aligned} \delta^o_{e,y,i} &= \alpha\delta_{e,y,i} + (1-\alpha)\delta^o_{e,y,i-1} \\ \delta^o_{e,x} &= 0 \end{aligned} \tag{19}$$

The limitations from this filtering are raised in Section III-E.

E. Results and Discussion

Results from the haptic virtual fixture are shown in Figure 6. The positions are based on the estimated end-effector position. Both force cancellation cases resulted in a decrease in the y forces, so the user was able to move more freely in those directions. The stiffness control should reduce the forces experienced in the y direction relative to the force feedback only case because the commanded stiffness was less than the device stiffness. While there is a minor decrease when stiffness control is added according to the root mean square of the force, most of the y forces remain. The y forces in both control cases correlate with the velocity and are therefore damping forces. While damping was not explicitly considered, we expect the material damping to be small. The primary source of the observed damping force is likely the filtering of the bias position which was necessary to stabilize the system.

The deviation in the x direction was similarly low in all three cases. The effective stiffness in the x direction was approximated by fitting a line to the x force and displacement data. The correlation between the force and position was weak in all cases, reflecting imperfections in the system and control. Nonetheless, the stiffness in the x direction was highest in the case in which stiffness control was added.

The damping required to stabilize the system limits the effectiveness of the force feedback and the stiffness control. The system instability comes from time delay from the low frequency loop and error between the model and the real system, which is amplified because the model is relied upon for both the *low rate* position estimation and the force and stiffness control. The required damping could be reduced by sensing the end-effector position directly, for example with a magnetic tracker, rather than using the estimation. Also, running the control loop (and in particular the state estimation) at a higher rate would improve stability. As with all haptic

control loops, computation time is critical to system stability and performance. The low rate loop in our implementation ran at 45 Hz on a laptop with an Intel Core i7-6820HQ processor with 16 GB RAM. The stiffness control component depends on matrices already computed for the force-only control and is a relatively small linear equation to solve. Thus, the stiffness component itself does not add a substantial computational load. Optimizing the structure and software of the control loop for speed will be even more important for larger models and stiffer haptic environments.

IV. Conclusion

In this paper we presented a method for mapping stiffness from actuator to end-effector for a general deformable robot modeled with FEM, and demonstrated its effectiveness on a deformable robot. While we derived the coupled compliance matrix using FEM, the stiffness projection between actuators and end-effectors is not dependent on this type of model. Thus, the stiffness projection could be applied directly to other models that provide such a coupled compliance matrix. We also presented an estimation technique to determine the end-effector position and force for a deformable backdrivable robot. Finally, we used the stiffness mapping to enhance a control loop for a robot with a haptic virtual fixture.

There are many avenues for future work in both applications and theory. As with rigid robots, controllable stiffness can be used to moderate the interaction between a deformable robot and its environment, which is especially useful for manipulation. The analysis is applicable to a variety of actuation methods, from DC motors to pneumatics, and could provide insight during the design phase. In this paper we considered only robots with equal numbers of actuators and effectors; cases of redundancy or underactuation could be considered in future work. Further analysis is especially needed for cases when the actuation compliance cannot be coupled, which limits the feasible output stiffness. Using a full dynamic model could improve the results for deformable haptic devices and would be important for robots moving at high speed. Another area to consider with regard to stiffness control is redundancy in the configuration. Choosing a configuration that approximates a given target compliance could help in cases where actuator stiffness is limited. This requires knowledge of the change of stiffness of the device as the configuration changes, rather than the linearized model used here. Integrating other sensing could improve the state estimation. Finally, further design of the presented deformable robot could improve its range of motion and force.

References

[1] D. Rus and M. T. Tolley, "Design, fabrication and control of soft robots," *Nature*, vol. 521, no. 7553, pp. 467–475, May 2015.

[2] H. B. Gilbert, D. C. Rucker, and R. J. Webster III, "Concentric tube robots: The state of the art and future directions," in *Robotics Research*. Springer, 2016, pp. 253–269.

[3] N. Hogan, "Impedance Control: An Approach to Manipulation: Part II Implementation," *Journal of Dynamic Systems, Measurement, and Control*, vol. 107, no. 1, pp. 8–16, 1985.

[4] H. Choi, W. Chung, and Y. Youm, "Stiffness analysis and control of redundant manipulators," in *IEEE International Conference on Robotics and Automation*, 1994, pp. 689–695.

[5] K. Koganezawa and S. Ban, "Stiffness control of antagonistically driven redundant D.O.F. manipulator," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 3, 2002, pp. 2280–2285.

[6] A. Ajoudani, N. G. Tsagarakis, and A. Bicchi, "On the role of robot configuration in Cartesian stiffness control," in *IEEE International Conference on Robotics and Automation*, May 2015, pp. 1010–1016.

[7] C. Gosselin, "Stiffness mapping for parallel manipulators," *IEEE Transactions on Robotics and Automation*, vol. 6, no. 3, pp. 377–382, Jun. 1990.

[8] A. Firouzeh and J. Paik, "Grasp Mode and Compliance Control of an Underactuated Origami Gripper Using Adjustable Stiffness Joints," *IEEE/ASME Transactions on Mechatronics*, vol. 22, no. 5, pp. 2165–2173, Oct. 2017.

[9] A. Albu-Schaffer and G. Hirzinger, "Cartesian impedance control techniques for torque controlled light-weight robots," in *IEEE International Conference on Robotics and Automation*, vol. 1, 2002, pp. 657–663.

[10] A. Albu-Schäffer, M. Fischer, G. Schreiber, F. Schoeppe, and G. Hirzinger, "Soft robotics: what Cartesian stiffness can obtain with passively compliant, uncoupled joints?" in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, vol. 4, 2004, pp. 3295–3301.

[11] F. Petit and A. Albu-Schäffer, "Cartesian impedance control for a variable stiffness robot arm," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sep. 2011, pp. 4180–4186.

[12] A. De Luca and F. Flacco, "Dynamic gravity cancellation in robots with flexible transmissions," in *49th IEEE Conference on Decision and Control*, Dec. 2010, pp. 288–295.

[13] R. B. Gillespie, T. Shin, F. Huang, and B. Trease, "Automated Characterization and Compensation for a Compliant Mechanism Haptic Device," *IEEE/ASME Transactions on Mechatronics*, vol. 13, no. 1, pp. 136–146, Feb. 2008.

[14] N. Herzig, R. Moreau, T. Redarce, F. Abry, and X. Brun, "Non linear position and closed loop stiffness control for a pneumatic actuated haptic interface: the BirthSIM," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Sep. 2015, pp. 1612–1618.

[15] M. Mahvash and P. E. Dupont, "Stiffness control of a continuum manipulator in contact with a soft environment," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct. 2010, pp. 863–870.

[16] C. Della Santina, R. K. Katzschmann, A. Bicchi, and D. Rus, "Dynamic control of soft robots interacting with the environment," in *IEEE International Conference on Soft Robotics*, 2018.

[17] S. Hauser, M. Robertson, A. Ijspeert, and J. Paik, "JammJoint: A Variable Stiffness Device Based on Granular Jamming for Wearable Joint Support," *IEEE Robotics and Automation Letters*, vol. 2, no. 2, pp. 849–855, Apr. 2017.

[18] A. Firouzeh, M. Salerno, and J. Paik, "Stiffness Control With Shape Memory Polymer in Underactuated Robotic Origamis," *IEEE Transactions on Robotics*, vol. 33, no. 4, pp. 765–777, Aug. 2017.

[19] C. Majidi and R. J. Wood, "Tunable elastic stiffness with microconfined magnetorheological domains at low magnetic field," *Applied Physics Letters*, vol. 97, no. 16, p. 164104, Oct. 2010.

[20] N. Usevitch, A. Okamura, and E. Hawkes, "APAM: Antagonistic Pneumatic Artificial Muscle," in *IEEE International Conference on Robotics and Automation*, May 2018.

[21] F. Largillière, E. Coevoet, M. Sanz-Lopez, L. Grisoni, and C. Duriez, "Stiffness rendering on soft tangible devices controlled through inverse FEM simulation," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct. 2016, pp. 5224–5229.

[22] C. Duriez, "Control of elastic soft robots based on real-time finite element method," in *IEEE International Conference on Robotics and Automation*, Karlsruhe, Germany, May 2013, pp. 3982–3987.

[23] Z. Zhang, J. Dequidt, A. Kruszewski, F. Largilliere, and C. Duriez, "Kinematic modeling and observer based control of soft robot using real-time Finite Element Method," in *IEEE/RSJ International Conference on Intelligent Robots and Systems*, Oct. 2016, pp. 5509–5514.

[24] E. Coevoet, T. Morales-Bieze, F. Largilliere, Z. Zhang, M. Thieffry, M. Sanz-Lopez, B. Carrez, D. Marchal, O. Goury, J. Dequidt *et al.*, "Software toolkit for modeling, simulation, and control of soft robots," *Advanced Robotics*, vol. 31, no. 22, pp. 1208–1224, 2017.

[25] J. S. Przemieniecki, *Theory of matrix structural analysis*. Courier Corporation, 1985.