The 2010 IEEE/RSJ International Conference on
Intelligent Robots and Systems
October 18-22, 2010, Taipei, Taiwan

# Model Based Deformable Object Manipulation Using Linear Robust Output Regulation

Richard Fanson and Alexandru Patriciu

*Abstract*— **This paper presents an approach to the indirect simultaneous positioning task of deformable objects based on robust linear output regulation methods. Indirect control requires maneuvering control points defined on a deformable body to desired locations by manipulating points located elsewhere on the object. The proposed control scheme uses a linearization of the deformable object dynamics into a state-space model for which the inputs are the forces applied to the manipulation points and the outputs are the positions of the control points. Then, the indirect simultaneous positioning task is treated as a classical robust linear output regulation problem. The proposed controller can compensate for deformable model nonlinearities and material uncertainties. The controller performance is illustrated through simulation and experimental results obtained on a planar deformable object model. The experiment was conducted using a robot controlled to apply desired end-effector forces on the planar object.**

## I. Introduction

Robot manipulation has been used in manufacturing and automated industries for several decades. Most of these applications pertain to manipulating objects that are assumed rigid. However, there are many applications where the ability to manipulate deformable objects with the precision of a robot manipulator would be beneficial. Examples could include assembly of rubber materials, textile manipulation, or soft tissue manipulation. Applications, such as soft tissue handling in medical applications, requires the control of the object's deformation, that is, the positioning of certain target points within the object. This type of interaction is commonly known as indirect simultaneous positioning [1]. The goal is to move certain control points defined within an object to desired locations by only manipulating other points located elsewhere in the object, such as at the boundary. An illustrative example would be a needle insertion task required to take a biopsy from a specific spot within a tissue. The target could move as the needle is inserted requiring several attempts for a sample. However, it is possible to act on the boundary of an object in order to *stabilize* the target on the needle path. Solutions for different variations of this problem were presented by Mallapragada et al. [2], Torabi et al. [3] and Smolen and Patriciu [4]. All three approaches use deformable object models to analyze the interaction between needle, tissue, and manipulated boundary points.

This work was supported in part by NSERC and CFI.

R. Fanson is with the Department of Electrical and Computer Engineering, McMaster University, Hamilton, Canada rtfanson@grads.ece.mcmaster.ca.

A. Patriciu is with the Department of Electrical and Computer Engineering, McMaster University, Hamilton, Canada patriciu@mail.ece.mcmaster.ca.

Deformable object manipulation uses computational models of the object in order to predict the changes in the physical object when it is subjected to external forces. Several approaches can be taken to model the objects, such as finite element methods (FEM), mesh-free methods, or mass-spring damper networks. There are already several articles that use different control strategies based on different models to achieve this goal. Wada et al [5] describe a PID control scheme that allows them to guide certain control points of a coarse spring model of a 2D fabric. They use the positional relationship between the manipulation points and the control points, and a PID controller on a prismatic manipulator to achieve these desired manipulation points. Shibata and Hirai [6] use a model that relates input forces to the desired control point positions and show simulation results on a 1D linear mass-spring-damper model. Smolen and Patriciu [7] use a reproducing kernel particle method (RKPM) to model the object. They use a control scheme based on the calculation of a Jacobian between the manipulation points and the control points at incremental states of deformation.

In this paper we propose a solution to the indirect simultaneous positioning problem based on the linearization of a deformable model that relates the forces on the manipulation points to the positions of the control points. The control law is derived from the linear robust output regulation of this linearized model. This formulation provides more insight into the solvability of the problem and in determining the controller gains to achieve regulation. Simulations and experiments are performed on a planar object with one manipulation point, one control point, and one of its edges fixed as an essential boundary. Since the proposed approach starts from a relatively accurate model of the deformable object and applies robust control tools to achieve the desired regulation it is expected the resulting controller will be able to handle material non-homogeneities and geometric nonlinearities.

The problem of indirect simultaneous control is described in more detail in Section II. Object modeling and linearization is described in Section III with the control derivation in Section IV. The control performance is illustrated through simulation and experimental results in V and VI, respectively. Discussion of the results and proposals for future work follow in Section VII.

## II. Indirect Simultaneous Positioning

In this section we summarize the problem of indirect simultaneous positioning [1] and introduce our approach to

solve this problem. For this paper, we will concern ourselves with 2-dimensional planar objects.

Consider the task in which an elastic object, EO, has to be deformed in a certain way. The final shape is defined by specifying desired locations for some points of EO labeled control points, $\boldsymbol{p}_c^i$ for $i = 1, \ldots, N_c$. The deformable body is manipulated by applying forces on some boundary points labeled manipulation points, $\boldsymbol{p}_m^j$ for $j = 1, \ldots, N_m$. It has been shown in [5] that the number of manipulation points must be at least the number of control points. That is, $N_m \geq N_c$. Note that this condition is necessary but not sufficient. The controllability properties of a deformable object are a function of object structure, object material properties, and control point and manipulation point locations. Therefore, those have to be analyzed for each deformable object model. A deformable object body illustrating the control points with target positions, and manipulation points with external forces is shown in Fig. 1.

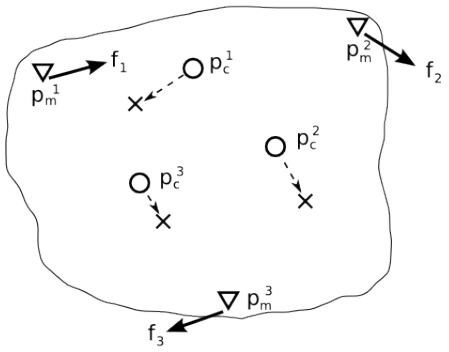


Fig. 1. Deformable object showing forces on manipulation points and control points being driven to target positions

The goal is to simultaneously drive the control points to their desired locations using only forces acting on the manipulation points. To achieve this task, our controller design uses the following steps:

1) model the deformable object using a suitable method such as FEM, mesh-free, mass-spring-damper, etc
2) linearize the model about the undeformed state to obtain a state-space representation
3) ensure the regulation solution exists by performing stabilizability, detectability, and solvability tests on the model
4) formulate a controller based upon linear robust output regulation theory

The modeling and linearization steps 1 and 2 are described in in Section III. The stabilizability, detectability, and solvability tests and controller formulation of steps 3 and 4 are discussed in detail in Section IV.

## III. Deformable Object Modeling and Linearization

Several different approaches can be used to model deformable objects, including finite-element methods (FEM), meshless methods, or spring-damper networks. Let's assume that the object is discretized into $N$ nodes. These should be such that some of them are located at manipulation points and the control points coordinates. There are also $N_n = (N - N_m - N_c)$ additional node points that are not control or manipulation points, $\boldsymbol{p}_n^k$ for $k = 1, \ldots, N_n$. Assuming a dynamic model, these discretized formulations can be shown, as in [8], to take the form

$$M\ddot{\boldsymbol{p}} + V\dot{\boldsymbol{p}} + K\boldsymbol{p} = \boldsymbol{F},$$

where $\boldsymbol{p} \in \mathbf{R}^N$ is a vector of displacements of each particle of the discretized model from its undeformed state. That is, $\boldsymbol{p}$ is a stacked vector containing positions of all points, $\boldsymbol{p}_c^i | i = 1 \ldots N_c$, $\boldsymbol{p}_m^j | j = 1 \ldots N_m$ and $\boldsymbol{p}_n^k | i = 1 \ldots N_n$, as they deviate from their original positions. Also, $M$ is an inertia matrix, $V$ is a damping matrix, $K$ is a stiffness matrix, and $\boldsymbol{F}$ is the external force vector. Depending on the type of modeling approach and the dimension of the problem, these matrices are likely a function of the position and velocity of the object nodes:

$$M(\boldsymbol{p})\ddot{\boldsymbol{p}} + V(\dot{\boldsymbol{p}}, \boldsymbol{p})\dot{\boldsymbol{p}} + K(\boldsymbol{p})\boldsymbol{p} = \boldsymbol{F}.$$

This second order dynamics equation can be reformulated to a set of first-order dynamics by introducing new states to represent both the displacements and velocities or each node in the model. That is, the state variable is $\boldsymbol{x} = \begin{bmatrix} \boldsymbol{p}^T & \dot{\boldsymbol{p}}^T \end{bmatrix}^T$. Also, the input control vector, $\boldsymbol{u}$, is obtained from decomposing the external forces such that $\boldsymbol{F} = \tilde{\boldsymbol{F}}\boldsymbol{u}$ where $\tilde{\boldsymbol{F}}$ is an $N \times N_m$ matrix of 1s and 0s indicating the states that represent manipulation points upon which the forces in $\boldsymbol{u}$ act. The nonlinear system is now described by

$$\begin{aligned} \dot{\boldsymbol{x}} &= f(\boldsymbol{x}(t), \boldsymbol{u}(t)) \\ \boldsymbol{y} &= h(\boldsymbol{x}(t)). \end{aligned} \tag{1}$$

Next, the system obtained by linearizing (1) around the origin is

$$\begin{aligned} \dot{\boldsymbol{x}} &= A\boldsymbol{x} + B\boldsymbol{u} \\ \boldsymbol{y} &= C\boldsymbol{x} \end{aligned} \tag{2}$$

where

$$A = \begin{bmatrix} 0 & I \\ -M^{-1}K & -M^{-1}V \end{bmatrix} \qquad B = \begin{bmatrix} 0 \\ M^{-1}\tilde{\boldsymbol{F}} \end{bmatrix}$$

evaluated at the undeformed operating point $(\boldsymbol{x}, \boldsymbol{u}) = (0, 0)$. Note that the output variables $y$ are just the positions of nodes that we have labeled as control points. As such, $C$ will just be a selector matrix that defines which states represent these control points, i.e. $\boldsymbol{y} = \begin{bmatrix} \boldsymbol{p}_c^{1\,T} & \ldots & \boldsymbol{p}_c^{N_c\,T} \end{bmatrix}^T$.

The indirect manipulation task is accomplished when outputs will have the desired values, $\boldsymbol{y}_d$. Formally, it is required to design $\boldsymbol{u}(t)$ such that

$$\lim_{t \to \infty} \boldsymbol{y}(t) = \boldsymbol{y}_d$$

Assuming the linearized model of the deformable object given by (2), one can recognize that the task at hand can be treated as an output regulation problem. The controller

should be robust with respect to deviations from the nominal plant due to nonlinearities and uncertainties inherent in the deformable object models. This is a classical problem in control systems and its solution is described in the next section.

## IV. Indirect Deformable Object Manipulation Using Output Regulation Control

The first complete treatment of the output regulation problem was presented by Francis [9]. Recently, a comprehensive treatment of output regulation for linear and nonlinear systems is presented by Huang [10]. For the completeness of the paper, this section summarizes the relevant concepts from these sources used to solve our output regulation problem.

### A. Robust Linear Output Regulation

The deformable object model represented by (2) is generalized to include exogenous signals, $\boldsymbol{v}$, that originate from some source outside the closed loop system. Also, we consider the error, $\boldsymbol{e} = \boldsymbol{y} - \boldsymbol{r}$, to be the output. Additionally, the nominal values of the plant matrices have uncertainty included such that the plant model is described by

$$\begin{aligned}\dot{\boldsymbol{x}} &= A_w\boldsymbol{x} + B_w\boldsymbol{u} + E_w\boldsymbol{v}\\ \boldsymbol{e} &= C_w\boldsymbol{x} + D_w\boldsymbol{u} + F_w\boldsymbol{v},\end{aligned}$$

where the subscript "$w$" indicates the state-space matrices deviate from their nominal values such that

$$\begin{aligned}A_w &= A + \Delta A, \quad B_w = B + \Delta B, \quad E_w = E + \Delta E\\ C_w &= C + \Delta C, \quad D_w = D + \Delta D, \quad F_w = F + \Delta F.\end{aligned}$$

The inclusion of $\boldsymbol{v} = \begin{bmatrix}\boldsymbol{r}^T & \boldsymbol{d}^T\end{bmatrix}^T$ terms indicate how the reference signals, $\boldsymbol{r}$, and disturbances $\boldsymbol{d}$, affect the plant states and outputs. The classical output regulation considers the case where exogenous signals, $\boldsymbol{v}$, are generated by $\dot{\boldsymbol{v}} = A_1\boldsymbol{v}$. However, for the task at hand we consider set-point inputs and assume no disturbance signals, making $\boldsymbol{v}$ constant and therefore $A_1 = 0$. In general, the effect of the exogenous signals on the system is incorporated by $E = \begin{bmatrix}0 & E_d\end{bmatrix}$, and $F = \begin{bmatrix}-I & F_d\end{bmatrix}$.

Following [10], if the full state is available, the control law can be used for robust output regulation is of the form

$$\begin{aligned}\boldsymbol{u} &= K_1\boldsymbol{x} + K_2\boldsymbol{z}\\ \dot{\boldsymbol{z}} &= G_1\boldsymbol{z} + G_2\boldsymbol{e}\end{aligned} \tag{3}$$

where $\boldsymbol{x}$ is the plant state vector, and $\boldsymbol{z}$ is a dynamic controller state vector to be defined later to ensure robustness. The closed loop system, with augmented state $\boldsymbol{x_c} = \begin{bmatrix}\boldsymbol{x}^T & \boldsymbol{z}^T\end{bmatrix}^T$, becomes,

$$\begin{aligned}\dot{\boldsymbol{x}}_c &= A_{cw}\boldsymbol{x}_c + B_{cw}\boldsymbol{v}\\ \boldsymbol{e} &= C_{cw}\boldsymbol{x}_c + D_{cw}\boldsymbol{v},\end{aligned}$$

where

$$A_{cw} = \begin{bmatrix}A_w + B_wK_1 & B_wK_2\\ G_2(C_w + D_wK_1) & G_1 + G_2D_wK_2\end{bmatrix},$$

$$B_{cw} = \begin{bmatrix}E_w\\ G_2F_w\end{bmatrix}, \quad C_{cw} = \begin{bmatrix}C_w + D_wK_1\end{bmatrix}, \quad D_{cw} = F_w.$$

The output regulation problem requires to design a controller such that The matrix $A_{C0}$ is Hurwitz, and

$$\lim_{t\to\infty}\boldsymbol{e}(t) = \lim_{t\to\infty}(C_{cw}\boldsymbol{x}_c(t) + D_{cw}\boldsymbol{v}(t)) = \boldsymbol{0} \tag{4}$$

Francis [9] showed that this is equivalent to finding the unique matrix $X_{cw}$ that solves the regulator equations:

$$\begin{aligned}X_{cw}A_1 &= A_{cw}X_{cw} + B_{cw}\\ 0 &= C_{cw}X_{cw} + F_{cw}.\end{aligned}$$

If the desired reference signals are constant then $X_{cw}A_1 = 0$.

The regulator equations are solvable if $(A, B)$ is stabilizable, $(C, A)$ is detectable, and

$$\operatorname{rank}\begin{bmatrix}A & B\\ C & D\end{bmatrix} = n + p, \tag{5}$$

where $n$ is the total number of states and $p$ is the number of outputs to be regulated.

Huang [10] shows that if the dynamic portion of the controller $\dot{\boldsymbol{z}} = G_1\boldsymbol{z} + G_2\boldsymbol{e}$ contains an internal model of the exogenous signals, then the pair

$$\left(\begin{bmatrix}A & 0\\ G_2C & G_1\end{bmatrix}, \begin{bmatrix}B\\ G_2D\end{bmatrix}\right)$$

is also stabilizable. In fact, this pair corresponds to the augmented system,

$$\begin{aligned}\dot{\boldsymbol{x}} &= A\boldsymbol{x} + B\boldsymbol{u} + E\boldsymbol{v}\\ \dot{\boldsymbol{z}} &= G_1\boldsymbol{z} + G_2\boldsymbol{e}\\ \boldsymbol{e} &= C\boldsymbol{x} + D\boldsymbol{u} + F\boldsymbol{v}.\end{aligned} \tag{6}$$

where the augmented plant matrices are

$$\tilde{A} = \begin{bmatrix}A & 0\\ G_2C & G_1\end{bmatrix}, \qquad \tilde{B} = \begin{bmatrix}B\\ G_2D\end{bmatrix}.$$

Then, a stabilizing controller for the augmented system is also a solution for the linear robust output regulation problem [10].

The controller (3) needs the full state and the desired output in order to perform the output regulation. However, it is unrealistic to assume that the full state is accessible since that would imply knowing every position and velocity of each node in the deformable model. A more realistic situation is one in which the only accessible information is composed of control point and manipulation point positions. The other positions and velocities must be estimated using a state estimator. This can be accomplished by including another dynamic state in the controller, $\hat{\boldsymbol{x}}$.

If we use a Luenberger observer, the new controller becomes

$$\begin{aligned}\boldsymbol{u} &= K_1\boldsymbol{x} + K_2\boldsymbol{z}\\ \dot{\boldsymbol{z}} &= G_1\boldsymbol{z} + G_2\boldsymbol{e}\\ \dot{\hat{\boldsymbol{x}}} &= A\hat{\boldsymbol{x}} + B\boldsymbol{u} + E\boldsymbol{v} + L\left(\boldsymbol{y}_m - C_m\hat{\boldsymbol{x}} - D_m\boldsymbol{u}\right)\end{aligned} \tag{7}$$

where $L$ is the observer gain matrix and $\hat{\boldsymbol{x}}$ is the estimation of our state variables. Notice that we have used the subscript

$m$ to differentiate between $\boldsymbol{y}_m$ and $\boldsymbol{y}$ for the cases when there is access to more states then just those we want to regulate. If we let $\boldsymbol{e_x} = \boldsymbol{x} - \hat{\boldsymbol{x}}$ represent the error between the true state and the observer state, we can see that

$$\begin{aligned}\dot{\boldsymbol{e}}_{\boldsymbol{x}} &= \dot{\boldsymbol{x}} - \dot{\hat{\boldsymbol{x}}} \\ &= (A - LC_m)\boldsymbol{e_x}.\end{aligned}$$

Thus we need to design $L$ such that $A - LC$ is Hurwitz to ensure the observer error asymptotically decreases to zero.

To show that this will still solve the output regulation problem we need to show that the closed loop system is stable. Let the closed loop state be $\begin{bmatrix}\boldsymbol{x} & \boldsymbol{z} & \boldsymbol{e_x}\end{bmatrix}$, where we will use the error of the observer, $\boldsymbol{e_x}$, instead of the observer state, $\hat{\boldsymbol{x}}$ because it gives a more useful realization. The closed loop system matrix $A_c$ becomes:

$$A_c = \begin{bmatrix} A + BK_1 & BK_2 & -BK_1 \\ G_2(C + D_K 1) & G_1 + G_2 DK_2 & -G_2 DK_1 \\ 0 & 0 & A - LC_m \end{bmatrix}$$

By the separation principle, the eigenvalues of the closed loop system are the union of the eigenvalues the augmented system (6) and $(A-LC_m)$. Since $K_1$, $K_2$ and $L$ are designed such that (6) and $(A-LC)$ are stable, the closed loop system is stable.

### B. Regulating Control Design Summary

In this section we summarize the output regulation control steps as it applies to our deformable object control. The overall process can be summarized as follows:

1) Model the object using the preferred method (e.g. FEM, mesh-free, etc.)
2) Linearize the model about the undeformed state to extract the state space model matrices $A$,$B$,$C$, and $C_m$. For these models, we assume $D$, $D_m$, and $E$ are zero, with $F$ simply $-I$.
3) Test if $(A, B)$ is stabilizable
4) Test if $(C, A)$ is detectable
5) Test if $\text{rank}\begin{bmatrix}A & B \\ C & 0\end{bmatrix} = n + p$
6) Find $G_1$ and $G_2$ such that $\dot{\boldsymbol{z}} = G_1 z + G_2 e$ incorporates an internal model of $A_1$. For constant reference signals and no disturbances it can be shown that setting $G_1 = 0_{p\times p}$ and $G_2 = I_{p\times p}$ incorporates an internal model.
7) Augment the system model, as in (6)
8) Find fixed gain matrix $K = \begin{bmatrix}K_1 & K_2\end{bmatrix}$ such that $\tilde{A} + \tilde{B}K$ is stable
9) Test if $(A^T, C^T)$ is stabilizable (if measured output $y_m$ is different from output to be regulated $y$)
10) Design observer gain matrix $L$ such that $(A - LC_m)$
11) Implement controller (7)

## V. Simulation

In this section, we will examine the robustness and performance of our control scheme for the indirect positioning of a deformable object first by simulation using MATLAB's Simulink, and then experimentally using a 5DOF robot manipulator.

### A. Mesh-free Deformable Object Model

For our simulations, we will consider a planar object as shown in Fig. 2. The object was defined to closely represent the phantom that would be used in the experiment, described in Section VI. The object is 140 mm in both length and width and discretized into a $9 \times 9$ grid with one edge fixed as an essential boundary. This restricts 9 of the nodes to be immobile, and thus are not included as states when we get to the state-space representation. The node at the center of the grid is used as the control point to be positioned to a target. Four nodes at the center of the opposite edge is where the external forces are applied. These are not independent manipulation points but rather, four points that will have the same control force input applied to them. The reason for this is to mimic how the robot end-effector contacts the phantom in the experimental setup. For our simulations,

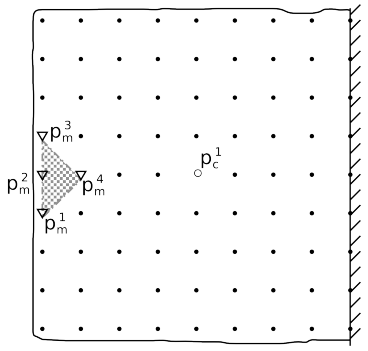


Fig. 2. Planar object implemented as a mesh-free RKPM model

the deformable object is modeled using a mesh-free particle method known as the reproducing kernel particle method (RKPM), similar to [7]. The output regulation controller is based on the linearization of this model. The control input signal determined by this controller is applied to the plant using this model to determine the deformation for the given applied forces. Since, the object modeling is not the focus of this paper, and for the sake of brevity, the foundations of this method will not be described here. The reader is referred to Chen et al. for more information [11]. The RKPM method is implemented in C++ using a CUDA acceleration to speed up the processing time.

For the state-space realization, we extract the form $M(\boldsymbol{d})\ddot{\boldsymbol{d}} + V(\dot{\boldsymbol{d}}, \boldsymbol{d})\dot{\boldsymbol{d}} + K(\boldsymbol{d})\boldsymbol{d} = \boldsymbol{F}$ from our model to get the $A$, $B$, $C$ and $D$ matrices after linearization. For this example, have 72 mobile nodes for which we have states of position and velocity in two dimensions and thus 288 states, i.e. $\boldsymbol{x} \in \mathbf{R}^{288}$. Also, we have a two dimensional control input, $\boldsymbol{u} \in \mathbf{R}^2$, and two dimensional output, $\boldsymbol{y} \in \mathbf{R}^2$, yielding:

$$\begin{aligned} A &\in \mathbf{R}^{288\times 288} & B &\in \mathbf{R}^{288\times 2} \\ C &\in \mathbf{R}^{2\times 288} & D &\in \mathbf{R}^{2\times 2}. \end{aligned}$$

To get better results from our estimator, we will use the manipulation points in addition to the control point as

measured output, $\boldsymbol{y_m} \in \mathbf{R}^{10}$, and

$$C_m \in \mathbf{R}^{10\times 288} \qquad D_m \in \mathbf{R}^{10\times 2}$$

although for our model, as mentioned before, both $D$ and $D_m$ are all zeros.

### B. Simulation Results

The simulation is conducted using MATLAB. The C++ RKPM model is compiled as a dynamic link library so that its functionality can be implemented as the plant model in Simulink simulations. The output of this plant, as well as the force measurements are given gaussian white noise with a variance of $0.01\,\mathrm{mm}^2$, and $0.001\,\mathrm{N}^2$, respectively, to simulate the sensor noise experienced the experimental setup.

We follow the steps outlined in Section IV-B to implement the controller. The gains for stabilizing the augmented plant are solved using linear-quadratic regulator (LQR) optimal control. Additionally, the stabilizing estimator gain matrix, $L$, is obtained from Kalman estimation techniques. The final controller is as described by (7)

We apply this control scheme to our system using a step reference signal of -14 mm in the x-direction and 4.2 mm in the y-direction. This allows us to see that the robustness can accommodate 10% of the body length in control point deformation. The resulting output for the control point defined at the center of the planar deformable object is shown in Fig. 3. For better visualization, Fig. 4 shows the trajectories of the control and manipulation points are overlaid with the initial and final deformations of the object. The results show the

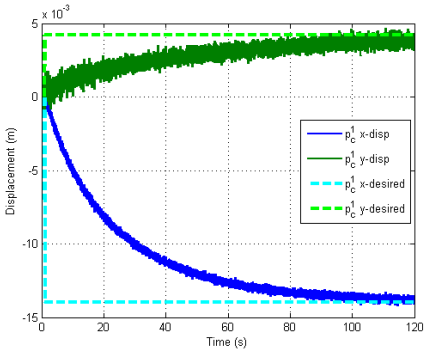


Fig. 3. Simulation output of the control point position.

the robust controller based on the linearization of our model achieves regulating even when the nonlinear plant deviates significantly from our linearization point.

## VI. Experiment

An experiment is conducted to test the robustness of the output regulation-based controller on an actual deformable phantom. The planar slab is modeled using the same mesh-free model with stress-strain parameters set to closely mimic the deformation of the actual object used. The forces are applied by a robot end-effector and the control point measured by a stereoscopic camera as described in the next section.

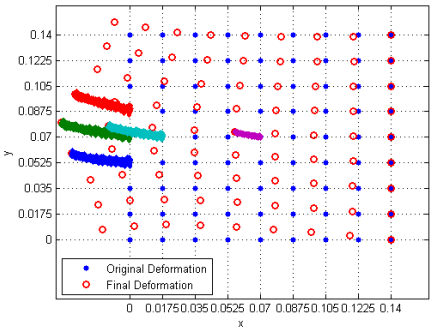


Fig. 4. Original and fully deformed object nodes with overlaid manipulation and control point trajectories.

### A. Experimental Setup

The planar slab is positioned on a table in the workspace of a robot with one edge clamped and resting atop ball bearings to reduce frictional effects during deformation. The robot manipulator is a 5DOF arm actuated by a hybrid position/force control scheme similar to that described in [12]. This ensures the end-effector orientation and z-coordinate (vertical) position are restricted by set values, while the motion in the x,y-plane is controlled to achieve a desired force. The actual force is measured from a force-torque sensor on the robot end-effector with the desired force set to be the control signal, $\boldsymbol{u}$, of the output regulation controller described in IV-B. The center of the object is marked using an infrared LED which is tracked in real-time using a stereoscopic camera. The x,y-displacement of the LED is extracted from the camera output and fed back into the controller as the output to be regulated, $\boldsymbol{y}$. Additionally, the robot end-effector displacement from its initial position when the object is undeformed, is fed back to the controller for the manipulation point displacements. Together with the control point, these displacements make up the measured output, $\boldsymbol{y}_m$. The experimental setup is seen in Fig. 5. Note that there is also coordinate frame registration aspect to this experiment to convert between camera and object coordinate systems. The camera is registered to the object using 3-pt registration from LEDs placed with known positions in the object frame.

### B. Experimental Results

The same control scheme parameters from the simulation used for the experiment. The reference signal was a step of -0.003 m in the x-direction and 0.002 m in the y-direction. The response of the system shown in Fig. 6. The robustness has clearly compensated for the actual object nonlinearities and fictional disturbances. The control signal force from the regulating controller as well as the measured force actually applied by the robot end-effector are plotted in Fig. 7.

Again we see the robustness compensating for the modeling errors as well as environmental disturbances.

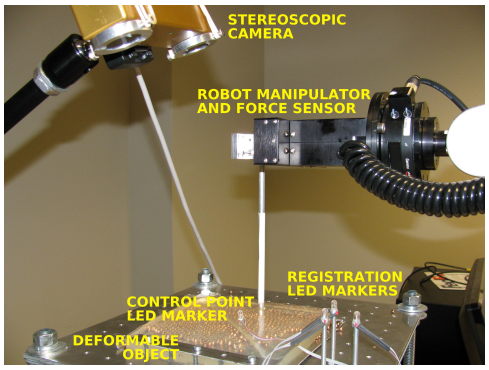


Fig. 5. Experimental setup showing object, manipulator, IR LED and camera

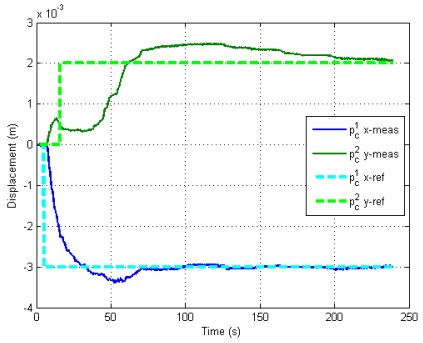


Fig. 6. Experiment output of the control point position.

## VII. Conclusion and Discussion

The indirect deformable object manipulation solution proposed in this paper comprises three steps. Firstly, a discretized model of the deformable object is built. Secondly, the discretized model is linearized around the origin allowing thus to formulate the indirect manipulation problem as an output regulation problem. Thirdly, a robust output regulator is synthesized for the linearized system. The nonlinearities and modeling errors are treated as disturbances to be compensated by the robust control scheme. Simulation and experiment illustrate the feasibility of the proposed approach.

For our simulations and experiments we use a planar object discretized into a $9 \times 9$ grid with one manipulation point and one control point. An RKPM mesh-free model relating the manipulation point forces to control point deformation is used as deformable object model. In the experiments, the forces are applied to a deformable phantom by a robot manipulator. The simulations and experiments both show that the controller can robustly position the control point to the desired position.

Our future work will focus on applying this control scheme to more complex objects, such as those with inhomogeneities. The proposed approach can be extended to include multiple control and manipulation points and 3-dimensional objects. One limitation of the current approach is that the dimension of the system can become large for complex deformable objects making the real time implementation challenging. In future we will address this issue by applying a model reduction technique [13] to the linearized system. Additionally, we will also explore the control of systems in which the robot dynamics is included into the plant model equations and have the robot motor torques as inputs. These models are valuable when the robot and deformable object dynamics properties are comparable.

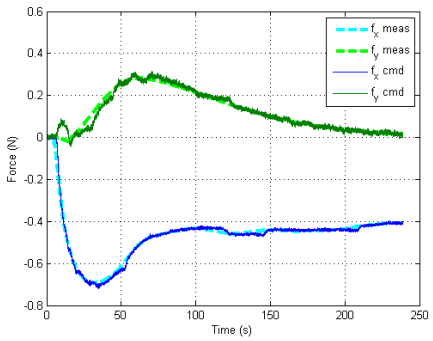


Fig. 7. Experiment force control input signals.

## References

[1] T. Wada, S. Hirai, and S. Kawamura, "Indirect simultaneous positioning operations of extensionallydeformable objects," in *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, vol. 2, 1998, pp. 1333–1338.

[2] N. P. T. Mallapragada, V.G. Sarkar, "Robot-assisted real-time tumor manipulation for breast biopsy," *IEEE Transactions on Robotics*, vol. 25, pp. 316–324, 2009.

[3] M. Torabi, K. Hauser, R. Alterovitz, V. Duindam, and K. Goldberg, "Guiding medical needles using single-point tissue manipulation," in *Proc IEEE Int. Conf. on Robotics and Automation*, May 2009, pp. 2705–2710.

[4] J. Smolen and A. Patriciu, "Model based stabilization of soft tissue targets in needle insertion procedures," in *Proc. Int. Conf. of the IEEE Engineering in Medicine and Biology Society*, Sept 2009.

[5] T. Wada, S. Hirai, S. Kawamura, and N. Kamiji, "Robust manipulation of deformable objects by a simple pid feedback," in *Proc. IEEE Int. Conf. on Robotics and Automation*, 2001, pp. 85– 90.

[6] M. Shibata and S. Hirai, "Soft object manipulation by simultaneous control of motion and deformation," in *Proc. IEEE Int. Conf. on Robotics and Automation*, 2006, pp. 2460– 2465.

[7] J. Smolen and A. Patriciu, "Deformation planning for robotic soft tissue manipulation," in *Proc. Int. Conf. on Advances in Computer-Human Interactions (ACHI)*, Feb. 2009, pp. 199–204.

[8] S. F. F. Gibson and B. Mirtich, "A survey of deformable modeling in computer graphics," MERL, Tech. Rep. TR-97-19, 1997.

[9] B. A. Francis, "The linear multivariable regulator problem," *SIAM Journal on Control and Optimization*, vol. 15, no. 3, pp. 486–505, 1977.

[10] J. Huang, *Nonlinear Output Regulation: Theory and Applications*. Philadelphia, PA: SIAM, 2004.

[11] J.-S. Chen, C. Pan, C.-T. Wu, and W. K. Liu, "Reproducing kernel particle methods for large deformation analysis of non-linear structures," *Computer Methods in Applied Mechanics and Engineering*, vol. 139, pp. 195–227, 1996.

[12] M. H. Raibert and J. J. Craig, "Hybrid position/force control of manipulators," *Journal of Dynamic Systems, Measurement, and Control*, vol. 103, no. 2, pp. 126–133, 1981.

[13] A. C. Antoulas, D. C. Sorensen, and S. Gugercin, "A survey of model reduction methods for large-scale systems," in *Structured Matrices in Mathematics, Computer Science, and Engineering I*, ser. Contemporary Mathematics. American Mathematical Society, 2001, vol. 280, pp. 193–219.