Visual Servoing Control of Soft Robots based on Finite Element Model

Zhongkai Zhang, Thor Morales Bieze, Jérémie Dequidt, Alexandre Kruszewski, Christian Duriez

Visual Servoing Control of Soft Robots based on Finite Element Model

Zhongkai Zhang, Thor Morales Bieze, Jeremie Dequidt, Alexandre Kruszewski, Christian Duriez

Abstract—In this paper, we propose a strategy for the control of soft robots with visual tracking and simulation-based predictor. A kinematic model of soft robots is obtained thanks to the Finite Element Method (FEM) computed in real-time. The FEM allows to obtain a prediction of the Jacobian matrix of the robot. This allows a first control of the robot, in the actuator space. Then, a second control strategy based on the feedback of infrared cameras is developed to obtain a correction of the effector position. The robust stability of this closed-loop system is obtained based on Lyapunov stability theory. Otherwise, to deal with the problem of image features (the marker points placed on the end effector of soft robot) loss, a switched control strategy is proposed to combine both the open-loop controller and the closed-loop controller. Finally, experiments on a parallel soft robot driven by four cables are conducted and show the effectiveness of these methods for the real-time control of soft robots.

Index Terms—Soft robot, Finite Element Model, simulation-based predictor, robust stability, image features loss

NOMENCLATURE

$\mathbf{x}$ Position vector of all FEM nodes.
$d\mathbf{x}$ Incremental displacement of nodes.
$\mathbf{f}_{ext}$ External load vector.
$\mathbf{f}(\mathbf{x})$ Internal stiffness forces vector.
$\mathbf{K}(\mathbf{x})$ Tangent stiffness matrix.
$\mathbf{H}_a$ Mapping matrix between nodes and actuator directions.
$\mathbf{H}_e$ Mapping matrix between nodes and effector directions.
λ_a Force contribution vector of actuators.
δ_a Position vector in the actuator space.
δ_e Position vector of effectors.
$\mathbf{W}_{ea}(\mathbf{x})$ Coupling matrix between effectors and actuators.
$\mathbf{W}_{aa}(\mathbf{x})$ Coupling matrix between actuators.
δ_e^{free} Position vector of effectors when $\lambda_a = 0$.
δ_a^{free} Position vector of actuators when $\lambda_a = 0$.
$\triangle\lambda_a$ Incremental force contribution of actuators.
$\triangle\delta_a$ Incremental position contribution of actuators.
$\mathbf{J}(\mathbf{x})$ Jacobian Matrix for soft robot.
$\hat{\mathbf{J}}(\hat{\mathbf{x}})$ The predicted Jacobian matrix.
$*^s$ Corresponding variables $*$ of the simulated FEM model.

I. INTRODUCTION

Compared to the rigid robots, soft robots are more suitable to be used in confined spaces [1], for manipulation of objects with different shapes [2], and for safe interaction with humans [3]. The promising applications stimulate researchers to develop effective methods for the design and analysis of soft robots. New design methodologies are proposed using promising soft actuator technologies [4], [5]. However, the accuracy and efficiency of soft robots is still limited by the difficulties of modeling and control of deformable systems that have, theoretically, an infinite number of degrees of freedom. With such a problem in mind, the purpose of this paper is to show a general kinematic model and tracking control method for soft robots.

The FEM provides an approach to model soft robots with generic assumption on the shape and on the constitutive material of the robots. The computation of FEM models is often considered as too costly for real-time applications. However, it has been proved that we are capable today of computing FEM models of soft tissues in real-time for haptic rendering [6]. In order to get the Jacobian matrix for the control design, a simulation-based predictor is realized by updating a simulation model built in SOFA [7] (an open-source real-time FEM simulator). In this paper, combining the soft robotic system and its simulation model, a closed-loop controller is designed and the condition of robust stability is provided when the error of the estimated Jacobian matrix is considered.

The closed-loop controller proposed in this paper employs the position-based visual servoing strategy. A fundamental requirement is to keep the image features available in the implementation of robot control. In general, image features loss can be caused by image processing error, camera breakdown, occlusions, or features leaving the camera field of view. One purpose of our work is to develop a more general method which can be widely used to solve this problem. Combining the open-loop and closed-loop control strategies, we present a novel switched control method to handle the image features loss. Compared with the switching strategy in [8] which tries to keep features in the field of view, our method can allow the image feature loss without constraint on the duration.

To the best of our knowledge, this is the first paper which addresses the robust stability of soft robots which are controlled based on the real-time FEM model. Three main contributions are shown in this paper: (1) A simulation-based predictor is designed to get the Jacobian matrix for both the open-loop and closed-loop control design of soft robot. (2) We obtain the condition of robust stability for the closed-loop system based on Lyapunov stability theory; (3) Combining the open-loop controller and the closed-loop controller, aswitched control method is proposed to deal with the problem of image features loss.

The authors are affiliated with INRIA, University of Lille, and Centrale Lille, France. Contact: zhongkai.zhang@inria.fr and christian.duriez@inria.fr

This paper is organized as follows. The related research in the field of model and control of soft robot is reviewed in Section 2. The discrete-time kinematic model and its features are shown in Section 3. In Section 4, The control problems about simulation-based predictor, open-loop control, and the closed-loop control are presented. Section 5 presents the switched control method to deal with the problem of image features loss. Some experiments are tested on a soft parallel robot and the results are presented in Section 6. Finally, conclusions and feature work are shown in Section 7.

II. RELATED RESEARCH

Several works about modeling and control of soft manipulators can be found in the literature. In [9], the visual servoing approach is applied for position control of cable-driven soft robotic manipulator based on a constant curvature models (CCM) model. In [10], a cascaded curvature controller is designed to control a highly compliant 2D manipulator actuated by bi-directional fluidic elastomer actuators. Then the method is extended to control a soft spatial fluidic elastomer manipulator [11]. In many practical cases, such as the significant contacts with the environment and the nonconstant deformation, the CCM does not capture all features of soft robots. Non-constant curvature models (NCCM) are therefore proposed, with a higher degree of accuracy, to model soft robots [12], [13]. However, controller design based on NCCM is more challenging. Besides, the traditional modeling methods are difficult to be used to model soft robots with complex geometrical shapes, like parallel deformable structures.

In the field of soft robotics, FEM is applied to model the deformation of soft structures [14], [15]. The first application of FEM in control design is shown in [16] where a novel method of modeling and control of soft robots is presented by extracting the reduced mechanical compliance in the space of the actuators and the end-effector. Further improvements and applications of this idea are presented in [17] where an asynchronous simulation framework is proposed to improve the control performances. Without using the feedback information from the robotic system, the above controllers are implemented in open-loop form which results in low control performances when the robots are modeled imprecisely. To overcome this drawback, in our former work [18], a closed-loop controller is designed with the assumption that the Jacobian matrix is estimated accurately. However, for real applications, the estimation error could affect the stability of the closed-loop system so that the robust stability is investigated in this paper.

Some existing methods that deal with image feature loss can be found. One strategy is to improve the algorithms to avoid any visual feature loss [19]. Another approach that addresses the problem of image feature loss is based on a prediction of the location of the features [20], [21]. Generally speaking, the methods based on prediction are available to continue the visual servoing task in case of feature loss. However, none of them can effectively deal with image feature loss during a long period of time.

III. BACKGROUND: KINEMATIC MODELING BASED ON REAL-TIME FINITE ELEMENT METHOD

In this section, the kinematic model of soft robot is deduced from the FEM model with the assumption of large displacements (non-linear internal forces) but ignoring inertia (we compute a succession of quasi-static problems) [16], [18], [22]. Then we show the basic features of the Jacobian matrix of soft robot.

A. Kinematic model

Based on FEM, the computation of soft robotic configurations is implemented step by step in the workspace of soft robots. The quasi-static equilibrium function of the entire robot at the $(k+1)th$ sampling time is given by:

$$\mathbf{f}_{ext,k+1} - \mathbf{f}(\mathbf{x}_{k+1}) + \mathbf{H}_{a,k+1}^T \lambda_{a,k+1} = 0 \qquad (1)$$

In (1), $\mathbf{f}(\mathbf{x}_{k+1})$ is expressed by its first-order Taylor expansion:

$$\mathbf{f}(\mathbf{x}_{k+1}) \approx \mathbf{f}(\mathbf{x}_k) + \mathbf{K}(\mathbf{x}_k) d\mathbf{x}_{k+1} \qquad (2)$$

Combining (1) and (2), the equilibrium equation of soft robots at each step can be established. To reduce its size, the equation can be projected into a constraint space defined by nodes that are the "effectors" of the robot and other nodes that are "actuated". See [17] or [18] for the detail:

$$\delta_{e,k+1} = \mathbf{W}_{ea}(\mathbf{x}_k) \lambda_{a,k} + \delta_{e,k}^{free} \qquad (3)$$

$$\delta_{a,k+1} = \mathbf{W}_{aa}(\mathbf{x}_k) \lambda_{a,k} + \delta_{a,k}^{free} \qquad (4)$$

where $\mathbf{W}_{ea}(\mathbf{x}_k) = \mathbf{H}_{e,k+1} \mathbf{K}^{-1}(\mathbf{x}_k) \mathbf{H}_{a,k+1}^T$ and $\mathbf{W}_{aa}(\mathbf{x}_k) = \mathbf{H}_{a,k+1} \mathbf{K}^{-1}(\mathbf{x}_k) \mathbf{H}_{a,k+1}^T$.

Assuming that the external force $\mathbf{f}_{ext}$ is constant (i.e. gravity), the kinematic equation (3) and (4) can be simplified as:

$$\delta_{e,k+1} = \delta_{e,k} + \mathbf{W}_{ea}(\mathbf{x}_k) \triangle \lambda_{a,k+1} \qquad (5)$$

$$\delta_{a,k+1} = \delta_{a,k} + \mathbf{W}_{aa}(\mathbf{x}_k) \triangle \lambda_{a,k+1} \qquad (6)$$

Then, combining (5) and (6), we have

$$\delta_{e,k+1} = \delta_{e,k} + \mathbf{J}(\mathbf{x}_k) \triangle \delta_{a,k} \qquad (7)$$

where $\mathbf{J}(\mathbf{x}_k) = \mathbf{W}_{ea}(\mathbf{x}_k) \mathbf{W}_{aa}^{-1}(\mathbf{x}_k)$.

At the next sampling time, the configuration of the robot is updated by

$$\mathbf{x}_{k+1} = \mathbf{x}_k + \mathbf{K}_k^{-1} \mathbf{H}_a^T \triangle \lambda_{a,k} \qquad (8)$$

so that the Jacobian Matrix $\mathbf{J}(\mathbf{x}_k)$ can be computed at each sampling time.

B. Features of the Jacobian Matrix

The features of $\mathbf{J}(\mathbf{x})$ depend on those of the compliance matrices $\mathbf{W}_{ea}(\mathbf{x})$ and $\mathbf{W}_{aa}(\mathbf{x})$. Two features needed for the controller design are:

(1) Non constant matrices. The compliance matrices $\mathbf{W}_{ea}(\mathbf{x})$ and $\mathbf{W}_{aa}(\mathbf{x})$ cannot be pre-computed because their values depend on the positions of all nodes (the configuration of soft robot) so that they change at each iteration. Based on the fact that the soft materials have slow dynamics and are not subject to high-frequency external loads, we make the assumption that the matrices $\mathbf{W}_{ea}(\mathbf{x})$ and $\mathbf{W}_{aa}(\mathbf{x})$ are constant between two sampling times.

(2) Rank of the matrix. The tangent stiffness matrix $\mathbf{K}(\mathbf{x})$ is positive definite when the robot is constrained to have no rigid body motion. By placing the actuators on different nodes of the FEM or with different directions, there will be no linear dependency between lines of $\mathbf{H}_a$ and $\mathbf{H}_e$ so that the matrices $\mathbf{W}_{aa}(\mathbf{x})$ is positive-definite and $\mathbf{W}_{ea}(\mathbf{x})$ is generally full row rank when the number of lines in $\mathbf{H}_e$ is smaller than the number of lines in $\mathbf{H}_a$, and $\mathbf{W}_{ea}(\mathbf{x})$ is full column rank in the opposite case. Further discussion about these matrices can be found in [18].

IV. CONTROL DESIGN FOR VISUAL SERVOING OF SOFT ROBOTS

The control strategy proposed in this paper consists of two parts: A predictor of the value of Jacobian matrix at each sampling time and a closed-loop controller for the tracking control of soft robot.

A. simulation-based predictor and open-loop control

For traditional robots, that have generally few Degrees of Freedom (DoF), Jacobian matrix $\mathbf{J}(\mathbf{x})$ can be computed easily using the state variables of the real robots. This is feasible because $\mathbf{J}(\mathbf{x})$ depends on some fixed parameters and several state variables which can be measured or observed on the robots directly. However, the way we obtain $\mathbf{J}(\mathbf{x})$ for soft robots, as presented above, is different. Namely, the Jacobian matrix only includes a small part of the state of the robots.

Thanks to the development of the real-time FEM technology, the soft robots can be simulated in real-time to get the value of $\mathbf{J}(\mathbf{x})$ at each sampling time after the configuration has been updated. A closed-loop controller for the simulation model can be designed based on (7) to make sure that the Jacobian matrix of the simulation model stays close to the Jacobian matrix of the real robot. Besides the function of estimating the Jacobian matrix, the simulation-based predictor can also generate an open-loop controller for soft robots.

As shown in Fig. 1, the implementation of open-loop controller consists of two parts: closed-loop control of simulation model and the open-loop control of soft robot. We still define the strategy as open-loop control because no feedback information from the soft robot is needed. In this strategy, both the simulation model and the soft robot share the same control input which is the contribution of the actuators and is computed based on the simulation model.

B. Closed-loop control of soft robots

The first consideration in the design of our closed-loop controller is the use of the simulation-based predictor. In
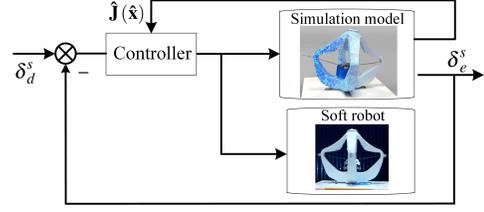

Fig. 1. The implementation of the open-loop controller. At each sampling time, $\hat{\mathbf{J}}(\hat{\mathbf{x}})$ is computed from the simulation model and is used for the computation of control input for both the simulation model and soft robot.
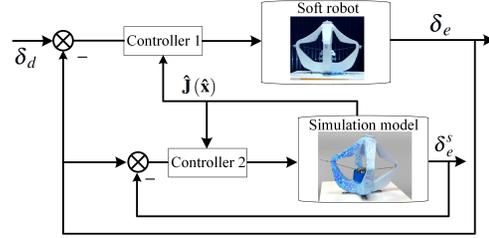

Fig. 2. The implementation of closed-loop controller. Controller 1 and Controller 2 are the closed-loop controllers for the soft robot and the simulation model, respectively. The two controllers can be designed independently.

order to decouple the controller design for the compound system, we propose the strategy to implement the closed-loop control which is illustrated in Fig. 2.

Controller 1 is designed to make sure that the end effector of soft robot tracks its desired trajectory δ_d. Controller 2 is employed to actuate the simulation model so that the end effector of simulation model follows the end effector of the real robot. Thanks to controller 2, the configurations of both simulation model and soft robot are similar. We assume that there is no big perturbation on the robot (strong contact or force) which would create dissimilar configurations between the real and simulated robot, while having a good registration of the effector (more formal proof of convergence with errors on $\mathbf{J}(\mathbf{x})$ can be found in section IV.D.). Therefore, the Jacobian matrix of soft robot can be predicted by the simulation model.

C. Closed-loop control law design

The control law design is based on proportional control strategy and pseudo-inverse control allocation. In order to simplify the controller design for trajectory tracking, two assumptions are required:

1. $rank \hat{\mathbf{J}}(\hat{\mathbf{x}}) = 3$ which means that there is no singular configuration in the workspace.

2. The actuators are not constrained (no saturation).

By defining a new control vector $U_k = \hat{\mathbf{J}}(\hat{\mathbf{x}}) \triangle \delta_{a,k}$, the kinematic model can be rewritten as

$$\delta_{e,k+1} = \delta_{e,k} + U_k \tag{9}$$

The tracking error is defined as $e_k = \delta_{e,k} - \delta_{d,k}$. In the task space, the control vector U_k can be designed as:

$$U_k = -k_p e_k - \left(\delta_{d,k} - \delta_{d,k+1} \right) \tag{10}$$

where k_p is a constant parameter for the proportional gain.

Then the Pseudo-inverse based control allocation is employed to obtain a unique solution:

$$u = \hat{\mathbf{J}}^+ (\hat{\mathbf{x}}_k) U_k \qquad (11)$$

which is the minimum 2-norm solution of $U_k = \hat{\mathbf{J}}(\hat{\mathbf{x}}_k) \triangle \delta_{a,k}$ and $u = \triangle \delta_{a,k}$ is the control input of the robot. For over-actuated system, the contribution of actuators with minimal energy can be computed by (11). Considering that the matrix $\hat{\mathbf{J}}(\hat{\mathbf{x}})$ has full row rank, the Pseudo-inverse matrix $\hat{\mathbf{J}}^+ (\hat{\mathbf{x}})$ can be computed by $\hat{\mathbf{J}}^+ (\hat{\mathbf{x}}) = \hat{\mathbf{J}}^T (\hat{\mathbf{x}}) \left(\hat{\mathbf{J}}(\hat{\mathbf{x}}) \hat{\mathbf{J}}^T (\hat{\mathbf{x}}) \right)^{-1}$.

Substituting (10) into the (11), the control law can be written as:

$$u = -\hat{\mathbf{J}}^+ (\hat{\mathbf{x}}_k) \left(k_p e_k + \delta_{d,k} - \delta_{d,k+1} \right) \qquad (12)$$

We use the same implementation strategy for the control law of controller 2, concerning the simulated robot:

$$u^s = -\hat{\mathbf{J}}^+ (\hat{\mathbf{x}}_k) \left(k_p^s e_k^s + \delta_{d,k}^s - \delta_{d,k+1}^s \right) \qquad (13)$$

where k_p^s is also a constant parameter for the proportional gain.

D. Closed loop stability analysis

In this subsection, we consider the situation where the Jacobian matrix is estimated imprecisely, i.e. $\hat{\mathbf{J}}(\hat{\mathbf{x}}) \neq \mathbf{J}(\mathbf{x})$. The object of the analysis is to get the range of control parameter k_p to make sure that the closed-loop system is stable. Considering that the closed-loop system is linear, the stability analysis is done with $\delta_{d,k} = \delta_{d,k+1}$.

Substituting (12) into (7), we have the closed-loop system:

$$e_{k+1} = (1 - k_p) e_k + \omega_k \qquad (14)$$

where

$$\omega_k = k_p \left[\mathbf{I} - \mathbf{J}(\mathbf{x}_k) \hat{\mathbf{J}}^+ (\hat{\mathbf{x}}_k) \right] e_k$$

We assume that, even if the Jacobian matrix is estimated imprecisely, the error is bounded. By introducing a scalar γ, we have $\omega_k^T \omega_k \leq \gamma^2 k_p^2 e_k^T e_k$ where $\gamma^2 \geq \| \left[\mathbf{I} - \mathbf{J}(\mathbf{x}_k) \hat{\mathbf{J}}^+ (\hat{\mathbf{x}}_k) \right]^T \left[\mathbf{I} - \mathbf{J}(\mathbf{x}_k) \hat{\mathbf{J}}^+ (\hat{\mathbf{x}}_k) \right] \|$ and $\gamma \geq 0$.

In order to keep the robust stability of the closed-loop system (14), we have the following theorem:

Theorem 1. *The robust stability of the closed-loop system (14) stands if*
$$k_p \in \left(0, \quad \frac{2}{1+\gamma} \right)$$

where $0 \leq \gamma < 1$.

Proof: We consider the Lyapunov canditate function:

$$V(e) = e^T e \qquad (15)$$

In the following, we will find the conditions so that the variation of Lyapunov function is negative:

$$
\begin{aligned}
\mathbf{d}V &\triangleq V(e_{k+1}) - V(e_k) \\
&= \mathbf{e}_{k+1}^T \mathbf{e}_{k+1} - \mathbf{e}_k^T \mathbf{e}_k \\
&= \left[(1-k_p)^2 - 1 \right] e_k^T e_k + 2(1-k_p) e_k^T \omega_k + \omega_k^T \omega_k \quad (16)
\end{aligned}
$$

In (16), we add the term $-\alpha \left[\omega_k^T \omega_k - \gamma^2 k_p^2 e_k^T e_k \right] + \alpha \left[\omega_k^T \omega_k - \gamma^2 k_p^2 e_k^T e_k \right]$ where $\alpha \geq 0$ is a accessorial parameter for the proof. Then we have

$$
\begin{aligned}
\mathbf{d}V &= \left[(1+\alpha\gamma^2) k_p^2 - 2k_p \right] e_k^T e_k + 2(1-k_p) e_k^T \omega_k \\
&\quad + (1-\alpha) \omega_k^T \omega_k + \alpha \left[\omega_k^T \omega_k - \gamma^2 k_p^2 e_k^T e_k \right] \quad (17)
\end{aligned}
$$

Given that $\omega_k^T \omega_k \leq \gamma^2 k_p^2 e_k^T e_k$, we can find a bound for $\mathbf{d}V$. (17) can be rewritten into a matrix form:

$$\mathbf{d}V \leq \left[\begin{array}{c} e_k \\ \omega_k \end{array} \right]^T \mathbf{B} \left[\begin{array}{c} e_k \\ \omega_k \end{array} \right] \qquad (18)$$

where

$$\mathbf{B} = \left[\begin{array}{cc} (1+\alpha\gamma^2) k_p^2 - 2k_p & 1-k_p \\ 1-k_p & 1-\alpha \end{array} \right]$$

To make sure that $\mathbf{d}V < 0$, the matrix $\mathbf{B}$ should be negative definite so that $1 - \alpha < 0$ or

$$(1+\alpha\gamma^2) k_p^2 - 2k_p < 0 \qquad (19)$$

and

$$\det(\mathbf{B}) = (\alpha\gamma^2 - \alpha - \alpha^2\gamma^2) k_p^2 + 2\alpha k_p - 1 > 0 \qquad (20)$$

The analytical solution of (19) and (20) is obtained from the following analysis.

From (19), we have $0 < k_p < \frac{2}{1+\alpha\gamma^2}$. In (20), $\alpha\gamma^2 - \alpha - \alpha^2\gamma^2 < 0$ with $\alpha > 1$ and $\gamma > 0$. In order to make sure that (20) has solution, we need the discriminant to be positive, i.e. $\alpha(1-\gamma^2)(\alpha-1) > 0$ so that $\gamma < 1$.

(20) can be rewritten in α as

$$-\gamma^2 k_p^2 \alpha^2 + k_p (k_p\gamma^2 - k_p + 2) \alpha - 1 > 0 \qquad (21)$$

The dominant term in (21) is negative. We need to have real roots, which means that the discriminant must be positive:

$$k_p^2 (\gamma^2 - 1) (\gamma^2 k_p^2 - k_p^2 + 4k_p - 4) > 0 \qquad (22)$$

With the condition of $k_p > 0$ and $\gamma < 1$, (22) can be simplified as

$$(\gamma^2 - 1) k_p^2 + 4k_p - 4 < 0 \qquad (23)$$

so that we have $k_p < \frac{2}{1+\gamma}$ or $k_p > -\frac{2}{\gamma-1}$. There is no intersection between $k_p < \frac{2}{1+\alpha\gamma^2}$ and $k_p > -\frac{2}{\gamma-1}$. Therefore, the solution of (23) is $k_p < \frac{2}{1+\gamma}$.

Now we check if there is intersection between the solution set of (21) and $\alpha > 1$. The solution of (21) is $\alpha_2 < \alpha < \alpha_1$ where α_1 and α_2 are the roots of the polynomial in (21). α_1 is computed as

$$\frac{\gamma^2 k_p - k_p + 2 + \sqrt{\gamma^4 k_p^2 - 2\gamma^2 k_p^2 + 4\gamma^2 k_p + k_p^2 - 4\gamma^2 - 4k_p + 4}}{2\gamma^2 k_p}$$

To make sure that (21) holds with $\alpha > 1$, we need $\alpha_1 > 1$, i.e.

$$f(\gamma, k_p) < \sqrt{\gamma^4 k_p^2 - 2\gamma^2 k_p^2 + 4\gamma^2 k_p + k_p^2 - 4\gamma^2 - 4k_p + 4} \tag{24}$$

where $f(\gamma, k_p) = \gamma^2 k_p + k_p - 2$. One can prove that there is no intersection between the solution set of (23) and $f(\gamma, k_p) > 0$. On the other hand, $f(\gamma, k_p) \leq 0$ implies $k_p \leq \frac{2}{\gamma^2+1}$ which is compatible with the constraint $k_p < \frac{2}{1+\gamma}$, given that $0 \leq \gamma < 1$. Therefore, we have $0 < k_p < \frac{2}{1+\gamma}$. ∎

V. SWITCHED CONTROL FOR IMAGE FEATURE LOSS

A common failure using visual servoing strategies is caused by the image feature loss when the feedback information is not available in applications. To deal with this problem, we introduce a switching control strategy. The switched controller consists of an open-loop controller, a closed-loop controller and switching conditions. If the image feature is available, the closed-loop controller shown in Fig. 2 is employed. When the image feature loss occurs, the controller switches to the open-loop strategy shown in Fig. 1.

At the last sampling time before the switching time, the contributions of the actuators computed by controller 1 and controller 2 (both are shown in Fig. 2) are δ_a and δ_a^s respectively. When the image loss occurs, the incremental contribution of the actuators is $\triangle\delta_a^s$ so that the control inputs for both systems are $\delta_a^s + \triangle\delta_a^s$. It is noted that the control inputs $(\delta_a^s + \triangle\delta_a^s)$ and δ_a for the soft robot at the switching time are different. To avoid abrupt motions at the switching time, the input $\delta_{a,k+1}$ of the soft robot should progressively switch from one input to the other:

$$\delta_{a,k+1} = \delta_{a,k} + p\left(\delta_{a,k}^s + \triangle\delta_{a,k}^s - \delta_{a,k}\right) \tag{25}$$

where p is a constant parameter ($0 < p < 1$). This parameter can be used to tune the length of the switching time period.

Remark 1. Intuitively, if the closed-loop subsystem is activated long enough with slow switching (i.e., long enough dwell or average dwell time), the energy increase caused by switching or unstable subsystem can be trade off to maintain the stability of the system [23]. For the implementation of the controller, we assume that the image loss happens occasionally so that switching frequency is low.

VI. EXPERIMENTS

In this section, we show the experimental setup and control performances for trajectory tracking of soft robot.

A. Experimental setup

The experimental setput for the visual servoing control of soft robot is depicted in Fig. 3. The system consists of a
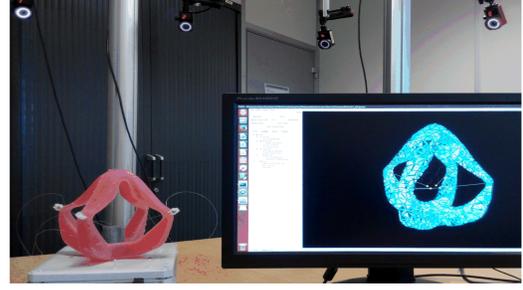

Fig. 3. Experimental setup for trajectory tracking of soft robot. The robot has four soft legs and is actuated by cables. The simulation model servers as a simulation-based predictor. The 3D position of the end effector can be obtained by the position perception system.

soft robot, a simulation model, and four cameras for position perception. For simplicity, we employ a commercial tracking system (*OptiTrack* by NaturalPoint company with sampling frequency 100Hz and a precision of 0.1mm) to track the 3-D position of end effector.

As a simulation-based predictor, the simulation model is built in SOFA framework and its configuration is updated with the frequency of 15 Hz. A volumetric mesh is generated based on the geometry of the robot. The mesh used for this study is composed by 4147 tetrahedra. The optimal number of elements in the mesh cannot be determined analytically, and it is usually chosen as a trade-off between accuracy and computation time. The soft robot is a parallel robot with four soft legs and is actuated by cables. Without using any rigid components, the robot is made of silicone with Young modulus being 240kPa. At each sampling time, the displacements of cables are computed by the control law and are sent to servomotors. In the experiments, the workspace of the robot is limited into a range where the forces of cables can be computed to be positive values.

B. Experiment results

The tracking performances with controllers proposed in this paper are shown in this section. Firstly, the performance to track a predefined circle trajectory with the closed-loop control strategy is illustrated. In this case, the position vector $\delta_{d,k+1}$ is known for controller 1 (see Fig. 2). However, it is unknown for controller 2 so that we simplify the controller 2 as $u^s = -k_p^s \hat{\mathbf{J}}^+(\hat{\mathbf{x}})e^s$. Then the switched control performance is presented in the case of image feature loss. The control strategy can also be employed for tracking other kinds of trajectory. The performances are similar so that we only show the performances for circle trajectory in this paper.

1) Control performances of closed-loop controller: The desired trajectory is predefined as a circle with radius 15 *mm* and the desired angular velocity is $2\pi/30$ (*rad/step*). We express the coordinates of the robot and of the simulation model in a unique coordinate system. To get the minimum tracking error, the optimal control parameter is tuned through intensive tests. In the experiments, the control parameters are chosen as $k_p = k_p^s = 0.5$.
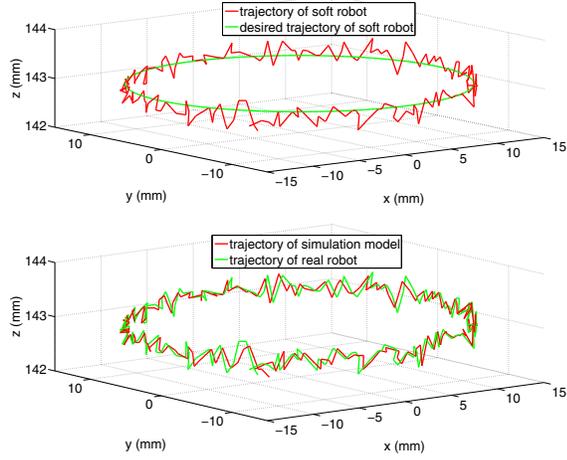
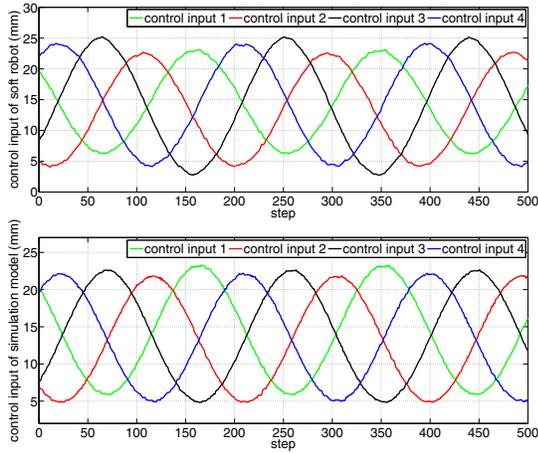Fig. 4. 3-D trajectories of real robot and simulation model using the closed-loop controller.

Fig. 5. Control inputs of soft robot and simulation mode using the closed-loop controller

Fig. 6. End effector trajectories of soft robot and simulation model using the switched control strategy.

The 3-D trajectory of end effector within one period (191 sampling times) is illustrated in Fig. 4. The average tracking errors for the real robot and the simulation model are 0.34 *mm* and 0.89 *mm* respectively. The synchronous movement can be guaranteed between the simulation model and the soft robot so that the Jacobian matrix can be predicted by the simulation model for the controller design.

The control inputs of the actuators are illustrated in Fig. (5). Because of the periodical movement of the end effector, the control inputs are also periodical. In order to compensate the unmodelled disturbances, chattering phenomenon is found in the control inputs.

2) Control performances of switched control: The control performances of switched control strategy are tested on the robot which is controlled to track the desired trajectory: a circle with radius 15 *mm* and angular velocity $2\pi/30$ (*rad/step*). When the image features are available, the closed-loop controller is activated with control parameters $k_p = k_p^s = 0.5$. The open-loop controller with $k_p^s = p = 0.5$ is employed
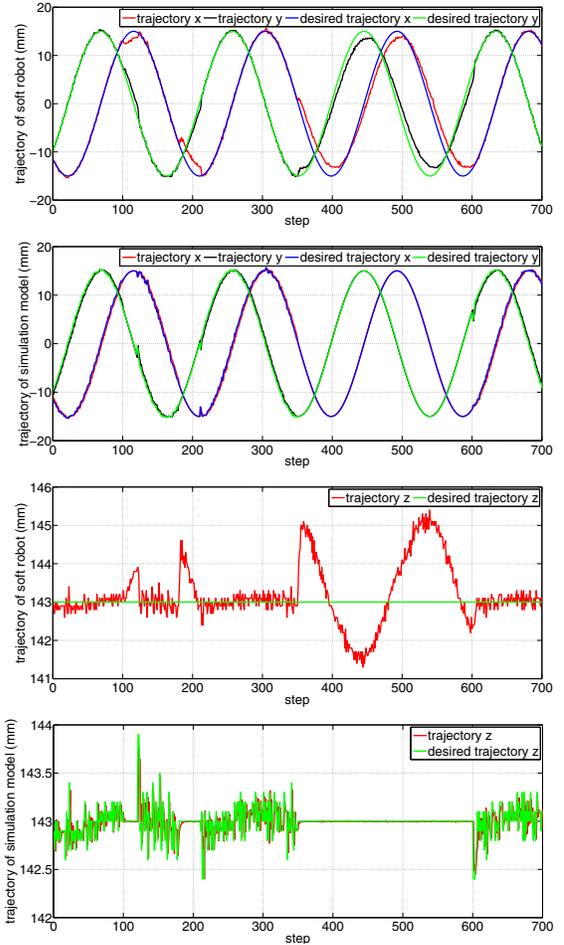
if the image features loss is detected. In the experiments, the image feature loss occurs between sampling steps 100-120, 180-220, and 350-600. The desired trajectory of simulation model is the trajectory of real robot. However, it changes to be the predefined trajectory when the image feature loss happens.

The trajectory of end effector is shown in Fig. 6. If the interval of image feature loss is shorter, the desired trajectory can be tracked with high accuracy using the switched controller. However, the tracking performance deteriorates when the interval is larger. It is noted that the performance of the proposed switch controller highly depends on the simulation model. Thanks to the high accuracy of finite element model, the tracking error is always limited in a small bound.

The components of control inputs are shown in Fig. 7. In the switching time, the chattering phenomenon becomes more significant which is similar to that of the trajectory. Without the feedback correction at each sampling time, the control inputs with open-loop controller are smooth, however at the cost of larger tracking error.
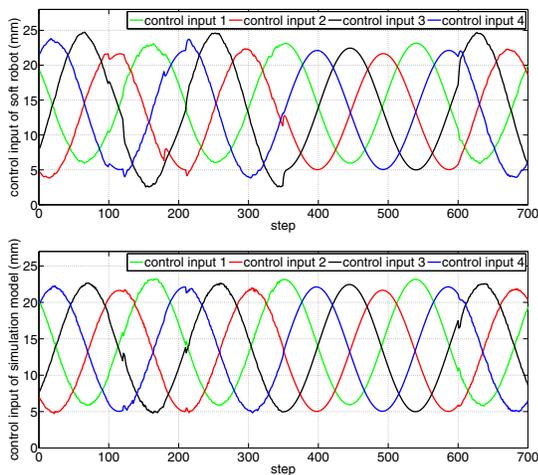
Fig. 7. Tracking control inputs of soft robot and simulation model using the switched control strategy.

VII. CONCLUSION

The paper presents a general method for the kinematic modeling and visual servoing control of soft robots based on FEM. The robot is simulated in real time to predict the Jacobian matrix and to generate an open-loop controller. Combined with the predictor and the soft robot, a closed-loop control strategy is proposed for visual servoing control of soft robot. Considering the unaccurate simulation model, the robust stability is proved based on Lyapunov stability theory. Futher, to deal with the problem of image feature loss during the visual servoing, the open-loop control and closed-loop control are combined to generate a switched strategy. A parallel soft robot actuated by cables is used to test these methods proposed in this paper.

The implementation of our strategy needs that the open-loop system is stable and the movement of the robot is quasi-static. In our further work, the constraints on the actuators and the workspace will be considered in the position control and path planning.

ACKNOWLEDGEMENT

REFERENCES

[1] R. F. Shepherd, F. Ilievski, W. Choi, S. A. Morin, A. A. Stokes, A. D. Mazzeo, X. Chen, M. Wang, and G. M. Whitesides, "Multigait soft robot," *Proceedings of the National Academy of Sciences*, vol. 108, no. 51, pp. 20400–20403, 2011.

[2] B. S. Homberg, R. K. Katzschmann, M. R. Dogar, and D. Rus, "Haptic identification of objects using a modular soft robotic gripper," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pp. 1698–1705, IEEE, 2015.

[3] S. Yim and M. Sitti, "Design and rolling locomotion of a magnetically actuated soft capsule endoscope," *Robotics, IEEE Transactions on*, vol. 28, no. 1, pp. 183–194, 2012.

[4] K. C. Galloway, P. Polygerinos, C. J. Walsh, and R. J. Wood, "Mechanically programmable bend radius for fiber-reinforced soft actuators," in *Advanced Robotics (ICAR), 2013 16th International Conference on*, pp. 1–6, IEEE, 2013.

[5] A. D. Marchese, C. D. Onal, and D. Rus, "Autonomous soft robotic fish capable of escape maneuvers using fluidic elastomer actuators," *Soft Robotics*, vol. 1, no. 1, pp. 75–87, 2014.

[6] H. Courtecuisse, Y. Adagolodjo, H. Delingette, and C. Duriez, "Haptic rendering of hyperelastic models with friction," in *Intelligent Robots and Systems (IROS), 2015 IEEE/RSJ International Conference on*, pp. 591–596, IEEE, 2015.

[7] F. Faure, C. Duriez, H. Delingette, J. Allard, B. Gilles, S. Marchesseau, H. Talbot, H. Courtecuisse, G. Bousquet, I. Peterlik, and others, "Sofa: A multi-model framework for interactive physical simulation," in *Soft Tissue Biomechanical Modeling for Computer Assisted Surgery*, pp. 283–321, Springer, 2012.

[8] G. Chesi, K. Hashimoto, D. Prattichizzo, and A. Vicino, "Keeping features in the field of view in eye-in-hand visual servoing: A switching approach," *IEEE Transactions on Robotics*, vol. 20, no. 5, pp. 908–914, 2004.

[9] H. Wang, W. Chen, X. Yu, T. Deng, X. Wang, and R. Pfeifer, "Visual servo control of cable-driven soft robotic manipulator," in *Intelligent Robots and Systems (IROS), 2013 IEEE/RSJ International Conference on*, pp. 57–62, IEEE, 2013.

[10] A. D. Marchese, K. Komorowski, C. D. Onal, and D. Rus, "Design and control of a soft and continuously deformable 2d robotic manipulation system," in *Robotics and Automation (ICRA), 2014 IEEE International Conference on*, pp. 2189–2196, IEEE, 2014.

[11] A. D. Marchese and D. Rus, "Design, kinematics, and control of a soft spatial fluidic elastomer manipulator," *The International Journal of Robotics Research*, p. 0278364915587925, 2015.

[12] D. Trivedi, A. Lotfi, and C. D. Rahn, "Geometrically exact models for soft robotic manipulators," *Robotics, IEEE Transactions on*, vol. 24, no. 4, pp. 773–780, 2008.

[13] F. Renda, M. Giorelli, M. Calisti, M. Cianchetti, and C. Laschi, "Dynamic model of a multibending soft robot arm driven by cables," *Robotics, IEEE Transactions on*, vol. 30, no. 5, pp. 1109–1122, 2014.

[14] N. W. Bartlett, M. T. Tolley, J. T. Overvelde, J. C. Weaver, B. Mosadegh, K. Bertoldi, G. M. Whitesides, and R. J. Wood, "A 3d-printed, functionally graded soft robot powered by combustion," *Science*, vol. 349, no. 6244, pp. 161–165, 2015.

[15] P. Polygerinos, Z. Wang, J. T. Overvelde, K. C. Galloway, R. J. Wood, K. Bertoldi, and C. J. Walsh, "Modeling of soft fiber-reinforced bending actuators," *Robotics, IEEE Transactions on*, vol. 31, no. 3, pp. 778–789, 2015.

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

[17] F. Largilliere, V. Verona, E. Coevoet, M. Sanz-Lopez, J. Dequidt, and C. Duriez, "Real-time control of soft-robots using asynchronous finite element modeling," in *Robotics and Automation (ICRA), 2015 IEEE International Conference on*, pp. 2550–2555, IEEE, 2015.

[18] 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 *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*, IEEE, Accepted, 2016.

[19] O. Kermorgant and F. Chaumette, "Dealing with constraints in sensor-based robot control," *IEEE Transactions on Robotics*, vol. 30, no. 1, pp. 244–257, 2014.

[20] D. Folio and V. Cadenat, "Dealing with visual features loss during a vision-based task for a mobile robot," *International Journal of Optomechatronics*, vol. 2, no. 3, pp. 185–204, 2008.

[21] N. Cazy, P.-B. Wieber, P. R. Giordano, and F. Chaumette, "Visual servoing when visual information is missing: Experimental comparison of visual feature prediction schemes," in *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 6031–6036, IEEE, 2015.

[22] C. Duriez, E. Coevoet, F. Largilliere, T. Morales-Bieze, Z. Zhang, M. Sanz-Lopez, B. Carrez, D. Marchal, O. Goury, and J. Dequidt, "Framework for online simulation of soft robots with optimization-based inverse model," in *Simulation, Modeling, and Programming for Autonomous Robots (SIMPAR), IEEE International Conference on*, pp. 111–118, IEEE, 2016.

[23] H. Lin and P. J. Antsaklis, "Stability and stabilizability of switched linear systems: a survey of recent results," *IEEE Transactions on Automatic control*, vol. 54, no. 2, pp. 308–322, 2009.