A Robust Model Predictive Controller for Tactile Servoing

Shuai Wang[1], *Senior Member, IEEE*, Yihao Huang[1,2], Wang Wei Lee[1], Tianliang Liu[1], Xiao Teng[1], Yu Zheng[1*], *Senior Member, IEEE*, Qiang Li[1,3*]

***Abstract*— Tactile servoing is an effective approach to enabling robots to safely interact with unknown environments. One of the core problems in tactile servoing is to robustly converge the contact features to the desired ones via a dedicated controller. This paper proposes a Data-Driven Model Predictive Controller (DDMPC) to compute the motion command given the previous interaction experience and feature deviations in tactile space. Compared with the manually designed PID-based controller, the proposed controller depends on the sound control theory and its convergence is guaranteed from a computational perspective. It is applied to the balancing control of a rolling bottle on a robotic forearm covered by a custom tactile sensor array. The real experiment demonstrates the superior robustness of the proposed approach and shows its great potential for other tactile servoing scenarios with measurement noise, which is inevitable for current tactile sensors.**

Fig. 1. Stability regulation of a bottle positioned on a robotic arm subject to manual disturbances. Tactile servoing is performed on a custom tactile sensor array covered by black foam on the forearm.

I. Introduction

Tactile sensors provide an important modality for intelligent robots to safely interact with unknown environments. They are attracting more and more attentions in robotics field. Researchers have already applied tactile sensing to unknown object exploration [1], classification [2], grasping [3], in-hand manipulation [4], walking [5] and human-robot interaction [6]. For all of these tasks, the core required technologies fall on three aspects: advanced tactile sensors, tactile perception, and tactile control. An early review of tactile sensors well explains different sensing principles (resisitive, capacitive, optical, magnatic, and etc.) and applications [7]. Recent years, we have also witnessed quick R&D progress of optical tactile sensors [8], which can provide rich and accurate contact information. The tactile perception and control as the tactile information are summarized from the computational opinion in [9]. Many important tactile features (tactile perception), e.g., contact position, pressure, curvature, and object mass, can be extracted from the raw tactile readout. Employing the extracted features, different control strategies (tactile control) can be designed for the robot to safely interact with unknown objects.

Tactile servoing is one research branch in tactile control. It uses feedback from tactile features to continuously control the pose of the robot to reach the desired goal features. In servoing tasks, action and perception are tightly coupled as the robot needs to continuously adapt its actions to the current tactile signals. Like visual servoing, tactile servoing also has two research lines: pose-based tactile tactile servoing and image-based tactile servoing.

Pose-based tactile servoing was mainly developed in the optical-based tactile sensor-tactip [10]. Along this line, the main challenge is to use the optical based tactile sensor to estimate the contacted 3-D tactile features, e.g., an edge. This requires to compute a mapping from high dimensional camera image to 3-D rotation and deep learning approach was tried and proved a feasible solution. Servoing scheme follows the traditional robotic Cartesian servoing control.

In image-based tactile servoing approaches [1], [11]–[14], the controlled features are defined in the tactile space and can be contact position, pressure and other contact geometries. The key of image-based tactile servoing is the computation of task Jacobian, which is the mapping from the twist motion of tactile sensor frame to the deviation in the tactile task space. Machine Learning (ML) approaches and model-based method were tried in computing the Jacobian. Along ML line, Wen et al. [12] proposed to use a deep neural network to represents the nonlinear relationship between current and desired pressure distributions and robot motion. Sutanto [14] divided the servoing control as two steps. At the first step, an offline neural-network-based manifold learning is performed to learn a latent space representation, which encodes the essence of the tactile sensing information. Secondly, a latent space dynamics model is learned from demonstration and deployed to perform an online control action computation based on both the current and target tactile images. Along

[1]The authors are with Tencent Robotics X, Shenzhen, Guangdong 518057, China.

[2]Y. Huang is also with the Department of Electrical & Electronic Engineering, School of Engineering, the University of Manchester, Manchester, M13 9PL, UK. This work is done when Huang works as an intern at Tencent Robotics X.

[3]Q. Li is also with the Department of Computer Science and Technology, Shenzhen Technology University. He was with Bielefeld University and participated in this work as a consultant for Tencent Robotics X.

*Co-corresponding authors (e-mails: petezheng@tencent.com, liqiang1@sztu.edu.cn)

the model-based line, our previous work on image-based tactile servoing [1] has shown the powerful capability in the object exploration, grasping [15], in-hand manipulation [16], and even tool-usage [17], although only simple image-based tactile features were used. In [1], a PID-type control strategy was used to compute the robot motion and reduce the tactile position deviation. This controller works well if the contacted object is not movable and the controller parameters are manually tuned to the optimal condition. Our experience shows that the manually designed tactile servoing controller cannot deal with dynamic and movable objects. One representative scenario is shown in Fig. 1. When the bottle on the sensitive forearm is disturbed and the robot needs to keep the contact position in the predefined position, position-based and manually designed controller cannot guide the tactile feature to desired one because of the limited sensor surface space. In this situation, we need an advanced controller which can use the high-order deviation of tactile features to predict future states.

One way to design a controller with high-order deviation of states is the model-based controller, such as the model predictive control (MPC). However, establishing the system model can be complex, or especially in the case that the model parameters vary with the environment or under disturbances. Motivated by human tactile afferents [18], we notice that phase of the past tactile data includes plenty of useful information about the system. Hence, proper application of the past tactile data will benefit tactile servoing. An easy option is to make the best use of past information and to make predictions of future states. Nowadays, collected past data is often used in robot control under plenty of data-driven approaches [19]–[21]. By combining the advantages of both past data and the prediction of the future states, Data-driven Model Prediction Control (DDMPC) [22] seems a promising option to be extended to the tactile domain. The DDMPC uses the past measured trajectories as an implicit model and thus does not require any prior identification step.

The main contributions of this paper include

- Exploiting the past tactile data, we propose a DDMPC approach for robust tactile servoing.
- Inspired by the general DDMPC theory, we extend it to tactile domain and prove its the stability.
- Beyond the theoretical analysis, we also verify the proposed controller in real experiment, in which the sensitive robotic arm is required to balance a rolling bottle on it.

The rest of this paper is organized as follows. Section II briefly presents in-house developed hardware, defines the tactile feature and analyzes sensor's measurement. Section III explains the DDMPC and proves its control stability. Section IV evaluates the controller and compares it with a manually designed controller. The experiments are shown in the accompanying video.

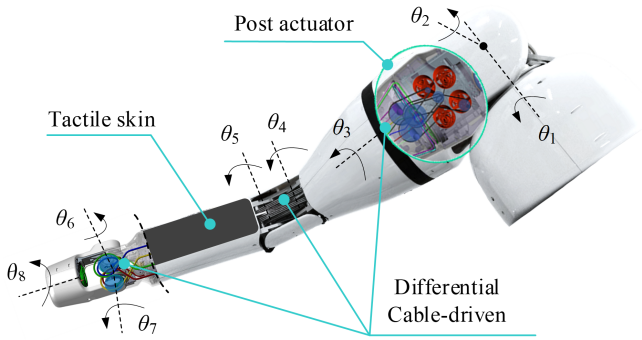


Fig. 2. TRX-Arm: a differential cable-driven robotic arm with a tactile sensor array covering its forearm. Both the arm and the tactile sensor array are developed by Tencent Robotics X.

II. Hardware and Sensor Evaluation

A. Robotic Arm

In this research, we use an in-house developed dexterous anthropomorphic arm (TRX-Arm) as a testbed, as shown in Fig. 2. It has 7 DoFs and is characterized by high dynamics with a maximum velocity of 7.4 m/s, a maximum acceleration of 44.5 m/s^2, and a payload of 6 kg as well as inherent compliance and safety. TRX-Arm has a similar configuration to that of the human arm, including 3 DoFs at the shoulder and the wrist as well as 1 DoF at the elbow. However, unlike traditional collaborative robotic arms, TRX-Arm adopts a novel remote actuation with differential cable transmission, which has the characteristics of low friction and inertia, self-compliance, and high dynamics.

A tactile sensor array is attached to the forearm, between joint angles θ_5 and θ_6. As for completing the tactile servoing task mentioned in this paper, global motion of the tactile sensor is computed via joint angles θ_1–θ_5, neglecting the posture and control of the wrist angles θ_6–θ_8. We denote by $\boldsymbol{\theta}$ the vector of all the joint angles.

B. Tactile Sensor

A custom tactile sensor array has been fabricated to cover the ventral section of the forearm. Based on the piezoresistive principle, the sensor array features a total of 706 tactels and forms a matrix shape. Figure 3 illustrates the essential tactels of the sensor. The electrodes and circuitry are first fabricated on a flexible Printed Circuit Board (FPCB), with tactels spaced 5 mm apart. On a separate sheet of Polyethylene Terephthalate (PET), patches of piezoresistive material are patterned using screen-printing. Adhesive is stencil printed around the piezoresistive patches. Subsequently, the PET sheet is glued onto the FPCB to form the sensor array. To enhance friction and compliance, a 2 mm layer of Ethylene-Vinyl Acetate (EVA) foam is pasted on top of the PET sheet, creating the exterior of the device.

As the sensor is a typical resistive array, many methods are available for attaining the resistances of each tactel [23]. Our readout method is an adapted version of the resistance matrix method described in [24]. Three separate microcontrollers

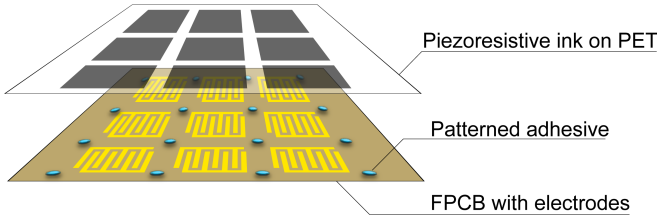


Fig. 3. Core components of the tactile sensor array.

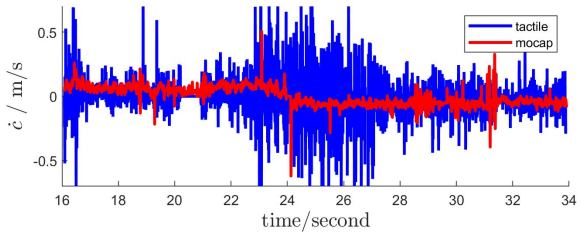


Fig. 4. An example of the tactile sensor velocity data $\dot{c}$.

are utilized to scan the entire array, achieving a readout speed of 300 Hz.

Since the sensor array's mounting onto the robot is performed by hand, positional errors are inevitable. 3D scanning is employed to capture the position and orientation of the sensor skin after installation, and a custom script is used to extract the position of individual tactels on the surface with respect to known fiducial markers on the robot itself.

C. Tactile Feature and Evaluation

Unlike the definition of tactile feature in [1] which refers to the contact position only, we augment the tactile velocity in the tactile feature to study the quick control response in this paper. First, the contact position is computed as the force-weighted center of pressure (CoP):

$$c = \frac{\sum_{ij \in R} f_{ij} c_{ij}}{\sum_{ij \in R} f_{ij}} \tag{1}$$

where f_{ij} and c_{ij} are the pressure and the discrete coordinate of the tactel in i-th row and j-th column, respectively. Due to the averaging effect from multiple tactels composing a contact region, we obtain a sub-tactel resolution for the contact position. Then, tactile velocity is defined as the first-order deviation of the contact position:

$$\dot{c} = \frac{c(t) - c(t-1)}{\Delta t}. \tag{2}$$

In summary, the tactile feature is defined as

$$\boldsymbol{y} = \begin{bmatrix} c & \dot{c} \end{bmatrix}^T. \tag{3}$$

We quantitatively evaluate the tactile velocity and find out that directly calculating the difference of CoP is quite noisy. Even using a high-order low-pass filter (i.e., 31st-order finite impulse response low-pass filter), the estimated tactile velocity is still not usable. To illustrate the noise, we compare the computed tactile velocity with the ground truth provided by motion capture in Fig. 4.

Directly using the noisy tactile position and velocity as the proportional and derivative components in a PID-type controller would result in high-frequency noise in the computed joint motion. To quickly respond to the change of the contact position, a high gain would be needed, which makes it more difficult to stabilize the tactile control. The proposed DDMPC in this paper can handle tactile measurements with such high noise.

III. DDMPC Tactile Servoing Controller

A. Definitions

The proposed tactile servoing controller is mainly used to replace the PID-type controller in the tactile servoing framework originally developed in [1]. The original PID-type controller computes the robotic joints motion to minimize the tactile features deviation. This controller requires the manually designed parameters and tends to fail in tactile exploration, especially when the robot needs to quickly respond to the change of the contact features. The DDMPC makes the control more robust by exploiting the past experience to predict future states.

Let $\boldsymbol{u}$ denote the inputs of a system and $\boldsymbol{y}$ the tactile feature measurement in the system. Let $(\boldsymbol{u}^s, \boldsymbol{y}^s) \in \mathbb{R}^{m+p}$ denote the steady state of the system with m inputs and p outputs. Let $(\bar{\boldsymbol{u}}, \bar{\boldsymbol{y}})$ denote the ideal trajectory of the system. Let $(\boldsymbol{u}^p, \boldsymbol{y}^p)$ denote offline collected past inputs and observable states data of the system. Usually, the collected tactile feature has noise. We use $\tilde{}$ to label data with noise. For example, offline noisy past data $(\boldsymbol{u}^p, \tilde{\boldsymbol{y}}^p)$ is collected before the online control by injecting an input time series $\boldsymbol{u}^p$ to the system and collecting the tactile feature sequence with noise $\tilde{\boldsymbol{y}}^p$. This noisy data is used as input to the online control. We consider output measurements with bounded additive noise in the initially available data. $\tilde{\boldsymbol{y}}^p_k = \boldsymbol{y}^p_k + \boldsymbol{\varepsilon}^p_k$ as well as in the online measurements $\tilde{\boldsymbol{y}}_k = \boldsymbol{y}_k + \boldsymbol{\varepsilon}_k$. We make no assumptions on the nature of the noise but require that it is bounded as $\|\boldsymbol{\varepsilon}^p_k\|_\infty \leq \bar{\varepsilon}$ and $\|\boldsymbol{\varepsilon}_k\|_\infty \leq \bar{\varepsilon}$ for some $\bar{\varepsilon} > 0$.

As shown in Fig. 5, apart from the above-mentioned steady states $(\boldsymbol{u}^s, \boldsymbol{y}^s)$ and the offline collected noise data $(\boldsymbol{u}^p, \tilde{\boldsymbol{y}}^p)$, the proposed controller also requires online past data measurements as input for the online feedback. We label the time index of a sequence in the subscript. For a stacked window of the sequence, we write

$$x_{[a,b]} = \begin{bmatrix} x_a & \cdots & x_b \end{bmatrix}^T. \tag{4}$$

We denote by x either the sequence itself or the stacked vector $x_{[0,N-1]}$ containing all of its components. Using the above terminology, online input measurements can be expressed as $(\boldsymbol{u}_{[t-n,t-1]}, \tilde{\boldsymbol{y}}_{[t-n,t-1]})$ and the control output is the actions $\boldsymbol{u}_{[t,t+n-1]}$ in a future time window, where the system is of order n with m inputs and p outputs.

B. Control Algorithm

The general idea of the control algorithm is summarized as follows. We use a matrix $\boldsymbol{h}(t)$ to store system dynamics

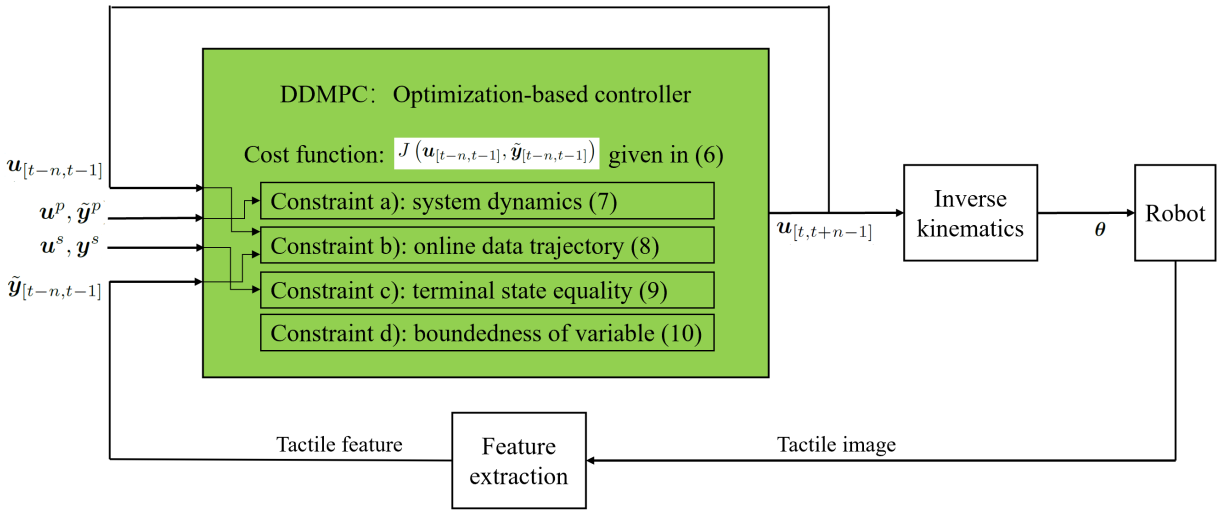


Fig. 5. DDMPC controller for tactile servoing.

information. Before the online control, system input and output data can be collected during an offline process. Then, instead of system identification, we use the offline collected data in the online control directly. The online controller can be formalized as an optimization problem. The system dynamics can be described in equality about the ideal trajectory $(\bar{\boldsymbol{u}}, \bar{\boldsymbol{y}})$, the offline collected data $(\boldsymbol{u}^p, \tilde{\boldsymbol{y}}^p)$, and the system dynamics matrix $\boldsymbol{h}(t)$. Similarly, the online collected data $(\boldsymbol{u}_{[t-n,t-1]}, \tilde{\boldsymbol{y}}_{[t-n,t-1]})$, as well as the equilibrium $(\boldsymbol{u}^s, \boldsymbol{y}^s)$ should also fit the ideal trajectory $(\bar{\boldsymbol{u}}, \bar{\boldsymbol{y}})$. These can also be described in equations. All these equations are used as constraints of the optimization. A slack variable $\boldsymbol{\sigma}$ is introduced to account for the noisy data, which can be interpreted as multiplicative model uncertainty.

In detail, we consider a quadratic stage cost, which penalizes the distance to a desired equilibrium $(\boldsymbol{u}^s, \boldsymbol{y}^s)$, i.e.,

$$l(\bar{\boldsymbol{u}}, \bar{\boldsymbol{y}}) = \|\bar{\boldsymbol{u}} - \boldsymbol{u}^s\|_{\boldsymbol{R}}^2 + \|\bar{\boldsymbol{y}} - \boldsymbol{y}^s\|_{\boldsymbol{Q}}^2 \tag{5}$$

where $\boldsymbol{Q}$ and $\boldsymbol{R}$ are positive-definite matrices.

We tackle the issue of noisy measurements in tactile servoing with a robust DDMPC scheme with terminal constraints. To account for noisy measurements, the relaxation parameter is penalized appropriately in the cost function. Given a noisy initial input-output trajectory $(\boldsymbol{u}_{[t-n,t-1]}, \tilde{\boldsymbol{y}}_{[t-n,t-1]})$ of length n, and noisy data $(\boldsymbol{u}^p, \tilde{\boldsymbol{y}}^p)$, the tactile servoing controller can be formalized as an optimization problem with the cost function

$$\begin{aligned} &J\left(\boldsymbol{u}_{[t-n,t-1]}, \tilde{\boldsymbol{y}}_{[t-n,t-1]}\right) \\ &= \min_{\substack{\boldsymbol{h}(t), \boldsymbol{\sigma}(t) \\ \bar{\boldsymbol{u}}(t), \bar{\boldsymbol{y}}(t)}} \sum_{k=0}^{L-1} \ell\left(\bar{\boldsymbol{u}}_k(t), \bar{\boldsymbol{y}}_k(t)\right) + \boldsymbol{\lambda}_h \bar{\boldsymbol{\varepsilon}} \|\boldsymbol{h}(t)\|_2^2 + \boldsymbol{\lambda}_\sigma \|\boldsymbol{\sigma}(t)\|_2^2 \end{aligned} \tag{6}$$

The optimization problem is also subjected to the following constraints:

(a) The constraint describing the system dynamics:

$$\begin{bmatrix} \bar{\boldsymbol{u}}_{[-n,L-1]}(t) \\ \bar{\boldsymbol{y}}_{[-n,L-1]}(t) + \boldsymbol{\sigma}(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{H}_{L+n}\left(\boldsymbol{u}^p\right) \\ \boldsymbol{H}_{L+n}\left(\tilde{\boldsymbol{y}}^p\right) \end{bmatrix} \boldsymbol{h}(t) \tag{7}$$

with the sequence $\{\boldsymbol{x}_k\}_{k=0}^{N-1}$ inducing the Hankel matrix

$$\boldsymbol{H}_L(\boldsymbol{x}) := \begin{bmatrix} \boldsymbol{x}_0 & \boldsymbol{x}_1 & \dots & \boldsymbol{x}_{N-L} \\ \boldsymbol{x}_1 & \boldsymbol{x}_2 & \dots & \boldsymbol{x}_{N-L+1} \\ \vdots & \vdots & \ddots & \vdots \\ \boldsymbol{x}_{L-1} & \boldsymbol{x}_L & \dots & \boldsymbol{x}_{N-1} \end{bmatrix}$$

where L is the prediction horizon of the MPC;

(b) The constraint ensuring that the internal state of the ideal trajectory aligns with the internal state of the predicted trajectory at time t:

$$\begin{bmatrix} \bar{\boldsymbol{u}}_{[-n,-1]}(t) \\ \bar{\boldsymbol{y}}_{[-n,-1]}(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{u}_{[t-n,t-1]} \\ \tilde{\boldsymbol{y}}_{[t-n,t-1]} \end{bmatrix}; \tag{8}$$

(c) The constraint specifying the terminal equality:

$$\begin{bmatrix} \bar{\boldsymbol{u}}_{[L-n,L-1]}(t) \\ \bar{\boldsymbol{y}}_{[L-n,L-1]}(t) \end{bmatrix} = \begin{bmatrix} \boldsymbol{u}_n^s \\ \boldsymbol{y}_n^s \end{bmatrix}, \quad \bar{\boldsymbol{u}}_k(t) \in \mathbb{U} \tag{9}$$

where $\mathbb{U}$ is the set of pointwise-in-time input constraints;

(d) Boundedness of the slack variable σ

$$\|\boldsymbol{\sigma}_k(t)\|_\infty \leq \bar{\boldsymbol{\varepsilon}}\left(1 + \|\boldsymbol{h}(t)\|_1\right), \quad k \in \mathbb{I}_{[0,L-1]} \tag{10}$$

where $\mathbb{I}_{[a,b]}$ is the set of integers in the interval $[a, b]$.

The output $\tilde{\boldsymbol{y}}^p$ and the initial output $\tilde{\boldsymbol{y}}_{[t-n,t-1]}$ obtained via online measurements are using measured noisy signal. The slack variable $\boldsymbol{\sigma}$ is added in the cost function to account for the noisy online measurements $\tilde{\boldsymbol{y}}_{[t-n,t-1]}$ and for the noisy data $\tilde{\boldsymbol{y}}^p$ used for prediction. The above ℓ_2-norm regularization for $\boldsymbol{h}(t)$ implies that small values of $\|\boldsymbol{h}(t)\|_2^2$ are preferred. The term $\boldsymbol{\lambda}_\sigma \|\boldsymbol{\sigma}(t)\|_2^2$ yields small values for the slack variable $\boldsymbol{\sigma}(t)$, thus improving the prediction accuracy.

The optimization result $(\bar{\boldsymbol{u}}(t), \bar{\boldsymbol{y}}(t))$ is an ideal trajectory of the system, so $\bar{\boldsymbol{u}}(t)$ can be used as the control actions. The application of the DDMPC-based tactile servoing approach is described in Algorithm 1.

Algorithm 1 DDMPC-Based Tactile Servoing Algorithm

1: Collect sufficient past output measurements with bounded additive noise in the initially available data $\boldsymbol{u}^p, \tilde{\boldsymbol{y}}^p$, with $\{\boldsymbol{u}_k\}_{k=0}^{N-1}$ being persistently exciting.
2: At time t, take the past n measurements $\boldsymbol{u}_{[t-n,t-1]}, \tilde{\boldsymbol{y}}_{[t-n,t-1]}$ and solve the optimization problem with the cost function (6) subject to the constraints (7)-(10).
3: Apply the input $\boldsymbol{u}_{[t,t+n-1]} = \bar{\boldsymbol{u}}^*_{[0,n-1]}(t)$ over the next n time steps.
4: Set $t = t + n$ and go back to 2).

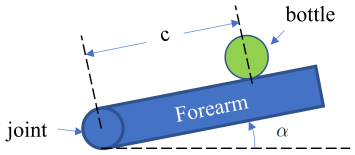


Fig. 6. Simplified planar model for the robotic forearm balancing a bottle.

C. Stability Analysis

For the convenience of stability analysis, we consider the following standard definition of persistence of excitation.

Definition 1: A sequence $\{\boldsymbol{u}_k\}_{k=0}^{N-1}$ with $\boldsymbol{u}_k \in \mathbb{R}^m$ is said to be persistently exciting of order L if $\operatorname{rank}(\boldsymbol{H}_L(\boldsymbol{u})) = mL$.

Our goal is to control an unknown linear time-invariant (LTI) system, denoted by $\boldsymbol{G}$, of order n with m inputs and p outputs, using only measured input-output data.

Definition 2: An input–output sequence $\{\boldsymbol{u}_k, \boldsymbol{y}_k\}_{k=0}^{N-1}$ is said to be a trajectory of an LTI system $\boldsymbol{G}$ if there exists an initial condition $\bar{\boldsymbol{x}} \in \mathbb{R}^n$ as well as $\{\boldsymbol{x}_k\}_{k=0}^{N}$ is a state sequence of a system $\boldsymbol{G}$ with a minimal realization of the state-space representation.

Definition 3: An input–output pair $(\boldsymbol{u}^s, \boldsymbol{y}^s) \in \mathbb{R}^{m+p}$ is said to be an equilibrium of an LTI system $\boldsymbol{G}$ if the sequence $\{\bar{\boldsymbol{u}}_k, \bar{\boldsymbol{y}}_k\}_{k=0}^{n}$ with $(\bar{\boldsymbol{u}}_k, \bar{\boldsymbol{y}}_k) = (\boldsymbol{u}^s, \boldsymbol{y}^s)$ for all $k \in \mathbb{I}_{[0,n]}$ is a trajectory of $\boldsymbol{G}$.

Lemma 1: [25] Suppose that $\{\boldsymbol{u}_k^p, \boldsymbol{y}_k^p\}_{k=0}^{N-1}$ is a trajectory of an LTI system $\boldsymbol{G}$, where $\boldsymbol{u}^p$ is persistently exciting of order $L + n$. Then, $\{\bar{\boldsymbol{u}}_k, \bar{\boldsymbol{y}}_k\}_{k=0}^{L-1}$ is a trajectory of $\boldsymbol{G}$ if and only if there exists $\boldsymbol{h} \in \mathbb{R}^{N-L+1}$ such that

$$\begin{bmatrix} \boldsymbol{H}_L(\boldsymbol{u}^p) \\ \boldsymbol{H}_L(\boldsymbol{y}^p) \end{bmatrix} \boldsymbol{h} = \begin{bmatrix} \bar{\boldsymbol{u}}_{[0,L-1]} \\ \bar{\boldsymbol{y}}_{[0,L-1]} \end{bmatrix}. \tag{11}$$

Lemma 1 lays the foundation for the DDMPC. It shows that a Hankel matrix, involving a single persistently exciting trajectory, spans the whole space of trajectories of an LTI system. It provides an appealing data-driven characterization of all trajectories of the unknown LTI system without requiring any prior identification step.

The nominal and robust version of DDMPC, with Lyapunov analysis to guarantee the closed-loop stability of the system, follows the spirit of [22]. The algorithm used in this paper follows the robust version because noise cannot be ignored in tactile servoing.

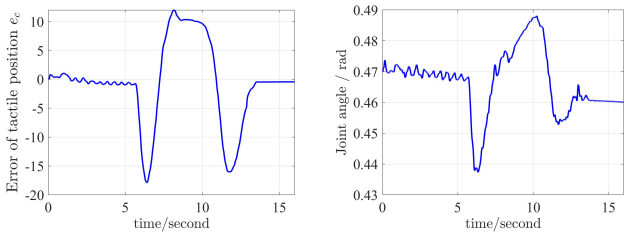
IV. Experimental Results

To demonstrate the effectiveness of the proposed DDMPC algorithm in tactile servoing, we conducted experiments in the case where the TRX-Arm equipped with the in-house developed tactile sensor array on the forearm is used to balance a cylindrical bottle that can roll on the forearm.

To start with, we tackle the balancing control with the feedback of position and velocity information of the contact between the bottle and the forearm. The system can be simplified as a planar model as depicted in Fig. 6. Let α represent the angle between the tactile sensor on the forearm and the horizontal plane, and c signifies the distance from the joint to the bottle. The tactile feature is defined as $\boldsymbol{y} = [c \;\; \dot{c}]^T$ with $\boldsymbol{u} = \alpha$ being the control input. Given the trajectory of α in the world frame, joint angle commands to control the robot can be calculated directly by inverse kinematics. In this case study, the dimension of matrices in the DDMPC can be calculated by setting $m = 1$, and $p = 2$. The prediction horizon is set to $L = 50$. Here c and $\dot{c}$ are obtained by tactile perception. The historically collected tactile perception data is used in the DDMPC-based tactile servoing controller. As mentioned in Sec. III, DDMPC is an optimization-based controller. The optimization problem in Algorithm 1 is solved online using qpOASES [26]. The control frequency remains consistent with the tactile sensor rate, which is 250 Hz. Although we only demonstrate the proposed controller for tactile servoing in one dimension in this example, it can be extended to high-dimension servoing easily using the same control theory if there are enough controllable degrees of freedom for the sensor.

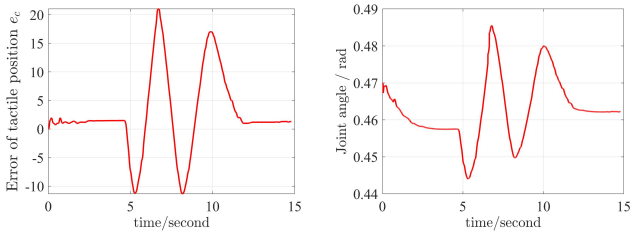
Before the online control, an offline past data $(\boldsymbol{u}^p, \tilde{\boldsymbol{y}}^p)$ collection process is required. Theoretically, the data can be collected without using any control policy. However, as a simple first attempt, we collected the past data with a length of $N = 200$ under a fine-tuning incremental PID controller. To ensure that the data captures enough information about the dynamics of the physical system, sufficient data both around and far away from the equilibrium points should be collected. For example, for each observable system state $\boldsymbol{y}$, data in different directions and varying trends should be gathered. This in turn guarantees that $(\boldsymbol{u}^p, \tilde{\boldsymbol{y}}^p)$ is persistently exciting of the system and can be used in Algorithm 1. For a piecewise LTI system, whose dynamics parameters are varying, it is also possible to update the past data $(\boldsymbol{u}^p, \tilde{\boldsymbol{y}}^p)$ online. This brings the benefits and potential of more flexible online adaptive control. However, the proofs and experiments for algorithms with online past data updates are beyond the scope of this paper.

The experiment used for comparison is using a fine-tuning incremental PID controller with high gain on the derivative term and saturation. In this tactile servoing task, velocity information is important for the balancing of the bottle, but unfortunately, the velocity cannot be measured accurately by the tactile sensor and is obtained by differentiation of the tactile position, so usually comes with high noise. This is a common problem in the tactile servoing. This bottle-



(a) Tactile sensor position error e_c (b) Robot joint angle

Fig. 7. Experimental data of fine tuning incremental PID controller for the bottle balancing task using tactile skin on the forearm.



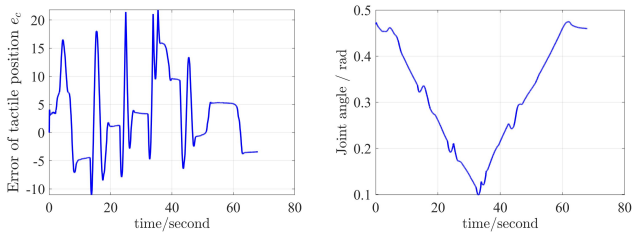
(a) Tactile sensor position error e_c (b) Robot joint angle

Fig. 8. Experimental data of the DDMPC for the bottle balancing task using tactile skin on the forearm.

balancing task can hardly be completed using traditional linear feedback control laws. To increase the speed of system response, high gain need be used to generate the control action. At the same time, the high gain should be within the joint motor's physical limits. After a few trials, a fine-tuning incremental PID controller with high gain on the derivative term and saturation managed to balance the bottle around the equilibrium. The system state and control input data are plotted in Fig. 7.

Following the idea proposed in this paper, the control results using DDMPC are shown in Fig. 8. It is clear that the robot joint angle command calculated using DDMPC is gentler as it makes the best use of past information and predicts future states. With this gentler controller, the robotic arm moved with less oscillation, which can be observed on the tactile position error e_c. Compared with traditional PID controllers, more past data, either online or offline, are included as state feedback. In addition, the structure of MPC enables control with a prediction of future horizons. However, the traditional PID control only uses the current state $\boldsymbol{y}_t$ and the past state $\boldsymbol{y}_{t-1}$ as feedback and no prediction of future states is made. Also, the weight of the bottle is not given in the DDMPC, as the past data contains the information of the whole system.

To test the robustness of the DDMPC-based controller, experiments were performed by giving the TRX-Arm a trajectory. Along the referenced trajectory, the forearm keeps parallel with the ground, making it possible to complete the forearm bottle balancing task at the same time. By adding the balancing control input on top of the referenced trajectory, the robotic arm followed the referenced trajectory and simultaneously kept the bottle balanced on the forearm



(a) Tactile sensor position error e_c (b) Robot joint angle

Fig. 9. Experimental data of the DDMPC for the bottle balancing with external disturbances for robustness testing.

without falling off. In this experiment, external manual disturbances are applied to the bottle at multiple time points, making the bottle roll on the forearm before settling down. The control results are shown in Fig. 9. Multiple disturbances injected into the system also demonstrate the robustness of the proposed controller.

V. Conclusions

Equipping robots with tactile sensors greatly expands their perception and adaptive control capabilities in the context of contacts. Advanced tactile sensors, perception, and control approaches are emerging in large numbers in recent years in the robotics domain. As an important control approach, tactile servoing has shown its capability in different scenarios, such as exploration, grasping, in-hand manipulation, and tool usage. In the current tactile servoing framework, the PID-type controller strongly depends on the manually designed parameters and cannot robustly guarantee convergence while robots interact with dynamic and movable objects. Aiming at this drawback, this paper formalizes the control as an optimization problem and exploits the interaction experience and online measured the tactile feature to propose a systematic designing approach called DDMPC. The stability of the proposed controller can be guaranteed in a weak tactile measurement assumption. The experimental results also show that the controller works well even with quite noisy measurement, which is common for most of the available tactile sensors. The advantage of the proposed approach over the PID-type controller has been demonstrated by a real experiment, which uses a robotic arm covered with a tactile sensor array to balance a rolling bottle.

As for the future work, we will consider system dynamics in tactile servoing. Experiments will be performed on different robotic platforms to explore how robot's locomotion and manipulation capabilities could be enhanced with the aid of tactile sensing.

Acknowledgment

The authors acknowledge Dr. Leilei Cui and Prof. Zhong-Ping Jiang for fruitful discussions.

References

[1] Q. Li, C. Schürmann, R. Haschke, and H. J. Ritter, "A control framework for tactile servoing." in *Robotics: Science and systems*. Citeseer, 2013.

[2] J. A. Fishel and G. E. Loeb, "Bayesian exploration for intelligent identification of textures," *Frontiers in Neurorobotics*, vol. 6, p. 4, 2012.

[3] M. Regoli, U. Pattacini, G. Metta, and L. Natale, "Hierarchical grasp controller using tactile feedback," in *Proceedings of the IEEE-RAS International Conference on Humanoid Robots*, 2016, pp. 387–394.

[4] H. van Hoof, T. Hermans, G. Neumann, and J. Peters, "Learning robot in-hand manipulation with tactile features," in *Proceedings of the IEEE-RAS International Conference on Humanoid Robots*, 2015, pp. 121–127.

[5] E. A. Stone, N. F. Lepora, and D. A. Barton, "Walking on tactip toes: A tactile sensing foot for walking robots," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2020, pp. 9869–9875.

[6] A. Cirillo, F. Ficuciello, C. Natale, S. Pirozzi, and L. Villani, "A conformable force/tactile skin for physical human–robot interaction," *IEEE Robotics and Automation Letters*, vol. 1, no. 1, pp. 41–48, 2015.

[7] R. S. Dahiya, G. Metta, M. Valle, and G. Sandini, "Tactile sensing—from humans to humanoids," *IEEE Transactions on Robotics*, vol. 26, no. 1, pp. 1–20, 2009.

[8] K. Shimonomura, "Tactile image sensors employing camera: A review," *Sensors*, vol. 19, no. 18, p. 3933, 2019.

[9] Q. Li, O. Kroemer, Z. Su, F. F. Veiga, M. Kaboli, and H. J. Ritter, "A review of tactile information: Perception and action through touch," *IEEE Transactions on Robotics*, vol. 36, no. 6, pp. 1619–1634, 2020.

[10] N. F. Lepora and J. Lloyd, "Pose-based tactile servoing: Controlled soft touch using deep learning," *IEEE Robotics & Automation Magazine*, vol. 28, no. 4, pp. 43–55, 2021.

[11] Z. Kappassov, J.-A. Corrales, and V. Perdereau, "Touch driven controller and tactile features for physical interactions," *Robotics and Autonomous Systems*, vol. 123, p. 103332, 2020.

[12] C.-T. Wen, S. Arai, J. Kinugawa, and K. Kosuge, "Tactile servoing based pressure distribution control of a manipulator using a convolutional neural network," *IEEE Access*, vol. 9, pp. 117 132–117 139, 2021.

[13] S. Tian, F. Ebert, D. Jayaraman, M. Mudigonda, C. Finn, R. Calandra, and S. Levine, "Manipulation by feel: Touch-based control with deep predictive models," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2019, pp. 818–824.

[14] G. Sutanto, N. Ratliff, B. Sundaralingam, Y. Chebotar, Z. Su, A. Handa, and D. Fox, "Learning latent space dynamics for tactile servoing," in *Proceedings of the IEEE International Conference on Robotics and Automation*, 2019, pp. 3622–3628.

[15] H. Liu, B. Huang, Q. Li, Y. Zheng, Y. Ling, W. Lee, Y. Liu, Y.-Y. Tsai, and C. Yang, "Multi-fingered tactile servoing for grasping adjustment under partial observation," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems)*, 2022, pp. 7781–7788.

[16] Q. Li, R. Haschke, B. Bolder, and H. Ritter, "Grasp point optimization by online exploration of unknown object surface," in *Proceedings of the IEEE-RAS International Conference on Humanoid Robots*, 2012, pp. 417–422.

[17] Q. Li, A. Ückermann, R. Haschke, and H. Ritter, "Estimating an articulated tool's kinematics via visuo-tactile based robotic interactive manipulation," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2018, pp. 6938–6944.

[18] R. S. Johansson and I. Birznieks, "First spikes in ensembles of human tactile afferents code complex spatial fingertip events," *Nature neuroscience*, vol. 7, no. 2, pp. 170–177, 2004.

[19] L. Cui, S. Wang, J. Zhang, D. Zhang, J. Lai, Y. Zheng, Z. Zhang, and Z.-P. Jiang, "Learning-based balance control of wheel-legged robots," *IEEE Robotics and Automation Letters*, vol. 6, no. 4, pp. 7667–7674, 2021.

[20] J. Zhang, S. Wang, H. Wang, J. Lai, Z. Bing, Y. Jiang, Y. Zheng, and Z. Zhang, "An adaptive approach to whole-body balance control of wheel-bipedal robot ollie," in *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2022, pp. 12 835–12 842.

[21] J. Zhang, Z. Li, S. Wang, Y. Dai, R. Zhang, J. Lai, D. Zhang, K. Chen, J. Hu, W. Gao *et al.*, "Adaptive optimal output regulation for wheel-legged robot ollie: A data-driven approach," *Frontiers in Neurorobotics*, vol. 16, p. 1102259, 2023.

[22] J. Berberich, J. Köhler, M. A. Müller, and F. Allgöwer, "Data-driven model predictive control with stability and robustness guarantees," *IEEE Transactions on Automatic Control*, vol. 66, no. 4, pp. 1702–1717, 2020.

[23] J.-f. Wu, "Scanning Approaches of 2-D Resistive Sensor Arrays : A Review," *IEEE Sensors Journal*, vol. 17, no. 4, pp. 914–925, 2016.

[24] L. Shu, X. Tao, and D. D. Feng, "A new approach for readout of resistive sensor arrays for wearable electronic applications," *IEEE Sensors Journal*, vol. 15, no. 1, pp. 442–452, 2014.

[25] J. C. Willems, P. Rapisarda, I. Markovsky, and B. L. De Moor, "A note on persistency of excitation," *Systems & Control Letters*, vol. 54, no. 4, pp. 325–329, 2005.

[26] H. J. Ferreau, C. Kirches, A. Potschka, H. G. Bock, and M. Diehl, "qpoases: A parametric active-set algorithm for quadratic programming," *Mathematical Programming Computation*, vol. 6, pp. 327–363, 2014.