# Neural-Network-Based Adaptive Model Predictive Control for a Flexure-Based Roll-to-Roll Contact Printing System

Jingyang Yan and Xian Du, *Member, IEEE*

***Abstract*—High-precision contact force control is essential for continuous roll-to-roll contact printing via mechanical contact on flexible web substrates using stamps. Nonuniformly controlled stamp contact force will cause failures during printing, especially for large-area printing processes. Due to their high precision in positioning and force control, flexure mechanisms have been applied in roll-to-roll contact printing systems; however, conventional physical model-based control systems cannot manage the nonlinear effects that exist in flexure-based roll-to-roll contact printing systems. To achieve precise contact force control, we propose a neural-network-based adaptive model predictive control for a flexure-based roll-to-roll contact printing system. The nonlinearity of the flexure mechanism is learned and modeled by an artificial neural network. To eliminate the steady-state error caused by model mismatches and external disturbances, an online adaptive mechanism is designed via updating the biases of the output layer of the neural network model. Experimental results show that the root-mean-square error of the contact force can be controlled in the range of 0–0.075 N with balances on two ends of the print roller, outperforming a proportional–integral–derivative controller, a neural-network-based standard model predictive control (MPC) controller, and a neural-network-based robust MPC controller. The proposed control algorithm is implemented in a microcontact printing process that prints 45-$\mu$m width gold patterns and achieves a variation of 0.3 $\mu$m in the average gold line width at different locations on an 88.9-mm width flexible substrate. The uniform microscale printing results have shown the effectiveness of the proposed neural-network-based adaptive model predictive control in the applied printing process.**

***Index Terms*—Adaptive model predictive control, contact force control, flexure mechanism, neural network (NN) modeling, roll-to-roll (R2R) contact printing.**

Manuscript received 17 October 2021; revised 2 February 2022 and 25 March 2022; accepted 1 May 2022. Date of publication 19 May 2022; date of current version 14 December 2022. Recommended by Technical Editor D. Dong and Senior Editor R. Gao. This work was supported by the NSF Foundation under Grant CMMI-1916866 and Grant CMMI-1942185. *(Corresponding author: Xian Du.)*

The authors are with the Mechanical and Industrial Engineering Department, Institute for Applied Life Sciences, University of Massachusetts, Amherst, MA 01003 USA (e-mail: jingyangyan@umass.edu; xiandu@umass.edu).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TMECH.2022.3172949.

Digital Object Identifier 10.1109/TMECH.2022.3172949

## I. Introduction

FLEXIBLE electronics have demonstrated potential in a wide range of applications in flexible displays, photovoltaics, medical devices, bio-integrated sensors, microfluidic devices, and computing platforms due to their lightweight, adaptability, and robustness. Roll-to-roll (R2R) printing methods, known for their high efficiency of mass production of flexible electronics, have been applied in the continuous printing of flexible electronics at low cost and high throughput [1]. Among these R2R printing methods, contact printing technologies, including microcontact printing [2], nanoimprinting [3], and offset printing [4], can transfer the pattern from a stamp to a substrate via ink contact and pressure. This relative simplicity of the printing process offers high scale-up potential for their applications in the mass production of flexible electronics. Due to the nature of the contact printing mechanism, the contact force is critical for the quality control of these R2R contact printing methods. For example, for microcontact printing in which an elastomeric stamp is used to transfer submicron scale patterns onto a flexible substrate via ink monolayers, over or imbalanced pressing of the stamp will cause failures of pattern transfer [5].

Conventional R2R contact printing control technologies are built with conventional mechanical components, which suffer from backlashes, assembly errors, and frictions; hence, preventing printing methods from achieving high-level precision [6], [7]. To overcome this problem, we have built a flexure-based R2R contact printing system as shown in Fig. 1, which consists of a web handling system and a controllable flexure-based printing module (refer to the detailed introduction of the test bench in Section III-A). Flexure mechanisms rely on bending and torsion of flexible elements, which allows us to achieve much higher precision in positioning compared to conventional mechanisms that rely on surface interaction between multiple moving parts [8]. Another key feature of the flexure mechanisms is that they are monolithic, which means there is no friction-induced wear and tear, and therefore no need for lubrication of components. Moreover, the monolithic build means complex assembly is not required, which altogether eliminates the possibility of assembly errors. Due to the superiority in motion and force control, flexures have been used in many R2R printing processes. For example, in [9], a five-axis flexure-based positioning system was introduced to compensate for misalignment in a multilayer R2R microcontact printing process. In [10], a

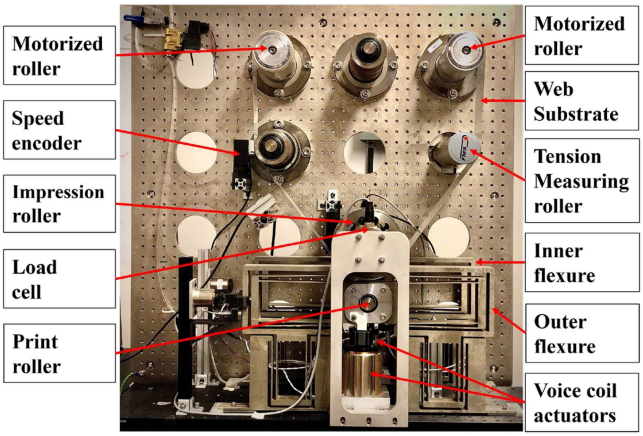


Fig. 1. Experimental setup setup of the flexure-based R2R contact printing system.

flexure-based microcontact printing platform was designed to achieve high-resolution printing. In [11], a flexure-based register system was designed for R2R electronics printings. The above-mentioned flexure-based R2R contact printing systems all used proportional–integral–derivative (PID) controllers for the force and position control based on their physical models [9]–[11]. However, the physical modeling of complex flexure mechanisms is challenging and time-consuming because many system parameters and inter-relationship among them need to be identified. For example, for the five-axis flexure mechanism in [9], up to 26 system parameters need to be identified. Meanwhile, the presence of nonlinearity in the flexure mechanisms makes it difficult to obtain an accurate physical model, causing the model mismatch problem and preventing the controller from providing correct inputs to the actuators. Although some classic adaptive and robust control methods can be used to solve the model mismatch problem, these methods still have difficulties in the system identification of the physical models of the complex flexure mechanisms as illustrated above. As a result, advanced control algorithms that are capable of dealing with nonlinear, time-varying dynamics of print processes need to be investigated further.

Fortunately, neural-network (NN)-based model predictive control (MPC) recently emerged with the intelligence capable of coping with the uncertainty present in the autonomous manufacturing process [12]. Such control has shown high performance with nonlinear, continuous, and differentiable activation functions. In the meantime, NN-based MPC has been widely used in precision motion control fields and has shown better performance than the conventional PID controller. For example, in [13], researchers demonstrated that the NN-based MPC can reduce the root-mean-square error (RMSE) by at least 50% compared with the PID control of piezo actuators for trajectory tracking. Also, MPC is a viable control strategy for a multiple-input and multiple-output system, such as flexure-based R2R contact printing, because of its intrinsic capabilities of handling multivariable interactions and constraints. Therefore, we propose an NN-based MPC for flexure-based R2R contact printing systems.

The accuracy of the NN-based MPC, however, can be diminished by temporary variations in the plant. In R2R contact printing systems, there are many disturbance sources including noncircularity, radius variations of the rollers, sliding of the web substrate, and variation of the web material properties. An NN-based MPC cannot be sufficiently and accurately trained offline to incorporate all temporary variations in the plant by limited data for online operation [14]. Meanwhile, external disturbances, such as temperature and humidity changes, always exist in industrial systems which could also induce prediction errors to the NN model [15]. In any case, a static NN-based MPC scheme would suffer from steady-state errors when facing model mismatches and external disturbances. In literature, there have been several ways to eliminate the steady-state errors for NN-based MPC methods. For example, external processes can be added to the controller to eliminate the steady-state error [16], [17]. In [16], an integrating controller is added in the outer loop around the controller. In [17], a disturbance model is trained to learn the external disturbances and model mismatches. The disadvantage of the above methods is that more complex structures are introduced to the controller. Alternatively, the trained NN model can be directly updated during each control loop to adapt to the disturbance. In [18], the entire NN model is updated during each control loop to compensate for the temporal mismatch between the online NN model and the plant. However, we find that the controller can become unstable if both weights and biases are updated during each control loop. The reason is that the dynamics of the system are stored in the weights and biases of the NN model. If all the weights and biases are updated by sequential sample data acquired from an online control loop, the NN model tends to forget the past subsequences' training information. Thus, the prediction error may occur for the new subsequent samples, leading to improper control action.

To overcome this problem, we propose a novel NN adaptive mechanism to eliminate the steady-state error, which only needs to update the biases of the output layer of the NN model to compensate for the error between the NN model output and the actual plant's output while still retaining the other weights' values for past sequences' training information. Our method adapts the biases to the model mismatches and external disturbances in dynamic processes, different from the method in [19] that only calculates prediction error in the steady-state and ignores the model mismatches and external disturbances in the dynamic processes. Our proposed method can update the NN model in every control loop and hence offers higher accuracy than [21] for the online control process.

The main contributions of this article are as follows:

1) A NN model is developed to characterize the dynamic behavior of the designed flexure mechanism in an R2R contact printing system.
2) A novel adaptive mechanism is designed for the NN-based MPC.
3) A prototype with successful mN-scale R2R contact force control using our NN-based MPC is implemented.

The rest of the article is organized as follows. The principles of the NN modeling and the adaptive MPC are described in Section II. The experimental setup, simulations, and comparative experimental results are shown in Section III. Finally, Section IV concludes this article.

## II. Neural-Network-Based Adaptive Model Predictive Control

### A. Neural Network Modeling

Many manufacturing processes are nonlinear which can be dealt with the nonlinear autoregressive exogenous model (NARX) as described by the following:

$$\mathbf{y}(k)=f(\mathbf{y}(k-1),\ldots,\mathbf{y}(k-n_y),\mathbf{u}(k-1),\ldots,\mathbf{u}(k-n_u)) \tag{1}$$

where $k$ denotes the $k$th sampling period, $f(\cdot)$ is an unknown nonlinear function, $\mathbf{u}\in\mathbb{R}^m$ denotes the control input series vector, $\mathbf{y}\in\mathbb{R}^n$ denotes the system output series vector, and $n_u$ and $n_y$ are, respectively, the exogenously designated time delays of the input and output. The NARX model can be implemented by using an NN architecture where feedback connections are enclosed between layers of the network. In this article, we model the NARX network using a single hidden layer feedforward NN to approximate the function $f(\cdot)$ by the following:

$$\mathbf{y_{NN}}(k)=\mathbf{w}_2\,\sigma(\mathbf{w}_1\mathbf{x}(k)+\mathbf{b}_1)+\mathbf{b}_2 \tag{2}$$

where $\mathbf{y_{NN}}(k)$ refers to the output of the NARX NN or prediction of the dynamic system for the $k$th period, $\mathbf{w}_1$ and $\mathbf{w}_2$ are, respectively, the weight matrices for the hidden layer and the output layer, $\mathbf{b}_1$ and $\mathbf{b}_2$ are, respectively, the bias vectors for the hidden layer and the output layer, and $\sigma(\cdot)$ is the activation function for the hidden layer. The input feature vector is defined as $\mathbf{x}(k)=[\mathbf{y}(k-1),\ldots,\mathbf{y}(k-n_y),\mathbf{u}(k-1),\ldots,\mathbf{u}(k-n_u)]$. According to the universal approximation theorem [20], the NN model can be trained to accurately model the nonlinear mapping $f(\cdot)$, i.e., $\|\mathbf{y_{NN}}(k)-\mathbf{y}(k)\|<\epsilon$ for any $\epsilon>0$ where $\epsilon$ is an arbitrarily small approximation bound. Therefore, the NN model can be trained by seeking the optimal solution to the following error function:

$$J_{\text{train}}=\sum_k\|\mathbf{y_{NN}}(k)-\mathbf{y}(k)\|^2. \tag{3}$$

This is accomplished by finding a set of weights and biases to minimize the error between the plant outputs and the predicted outputs of the NN.

### B. Neural-Network-Based Model Predictive Control

MPC calculates control commands by minimizing a cost function over a finite prediction horizon. The cost function is typically formulated by minimizing not only the squared error between the reference signal and the plant's output but also the weighted squared change of the control input. Given the NN architectures [refer to (2) and (3)] for the plant dynamics model, the MPC cost function can be formulated by the following:

$$J_{\text{MPC}}=\sum_{j=1}^{P}\|\mathbf{y_r}(k+j)-\mathbf{y_{NN}}(k+j)\|^2+\sum_{j=1}^{M}\lambda(j)\|\Delta\mathbf{u}(k+j)\|^2 \tag{4}$$

where $P$ is the prediction horizon, $M$ is the control horizon, $\mathbf{y_r}$ is the reference signal, $\mathbf{y_{NN}}(k+j)$ is the predicted output of the system at the $(k+j)$th period based on the available measurements at the $k$th sampling period, $\lambda(j)$ is the regulating factor for the control input that maintains the smoothness of the control signal and is typically chosen to be constant or exponential, and $\Delta\mathbf{u}(k+j)=\mathbf{u}(k+j)-\mathbf{u}(k+j-1)$ denotes the change in $\mathbf{u}$.

In the cost function (4), three parameters are required to be tuned, namely $P$, $M$, and $\lambda(j)$. The selection of control horizon $P$ and prediction horizon $M$ are correlated with both control performance and time consumption. For optimal performance, the prediction horizon $P$ should be sufficiently large compared with the settling time of the plant to achieve stability [21]. The weighting factors $\lambda(j)$ are used to control the balance between the two summations and acts as a damper to the control input. Smaller $\lambda(j)$ would result in more aggressive control moves. For simplicity, we chose $\lambda(j)$ to be constant.

The objective of the MPC is to minimize the cost function $J_{\text{MPC}}$ in (4) with respect to $[\mathbf{u}(k+1),\mathbf{u}(k+2),\ldots,\mathbf{u}(k+M)]$, denoted by $\mathbf{U}$. Since the plant model is an NN structure, a gradient descent method can be used to minimize the cost function iteratively. In each iteration, the intermediate values for $J_{\text{MPC}}$ can be denoted by $J_{\text{MPC}}(n)$. The control command sequence $\mathbf{U}(n)$ can be updated by the following:

$$\mathbf{U}(n+1)=\mathbf{U}(n)+\Delta\mathbf{U}(n) \tag{5}$$

$$\Delta\mathbf{U}(n)=\eta\left(-\frac{\partial J_{MPC}(n)}{\partial\mathbf{U}(n)}\right) \tag{6}$$

where $\eta>0$ is the update rate for a control command sequence. Since the two summation terms in the cost function are both functions of $\mathbf{U}$ modeled by the NN, it is feasible to calculate the Jacobian matrix $(\partial J_{\text{MPC}}/\partial\mathbf{U})$ through the backpropagation algorithm, which is the key process of the MPC. After $\mathbf{U}$ is optimized through the iterative process, only the first element of $\mathbf{U}$ is used as the input to the system.

### C. Eliminating Steady-State Error

The controller described in Section II-B relies on a static NN model, which suffers from model mismatches and external disturbances, and thus will have steady-state errors when applied in practical printing processes. To eliminate the steady-state error, a novel adaptive mechanism is proposed for the NN-based MPC model. First, an NN model is trained offline using training data from the plant. Then, the pretrained NN model is applied in the MPC controller to predict the plant outputs. According to the predicted outputs, an iterative process is carried out to find the optimal control input sequence. Next, the first element of the optimized control input sequence is sent both to the NN model and the real plant. Finally, the prediction error of the NN model is added to the biases of the output layer. The overall adaptive mechanism is elaborated upon in the pseudocode displayed in Algorithm 1. We prove below that the proposed adaptive algorithm can eliminate the steady-state error.

The steady-state error can be measured by the difference between the measured system output $\mathbf{y}(k)$ and the predicted

**Algorithm 1:** The Overall NN-Based Adaptive MPC Scheme.

**Input:** control horizon $M$, prediction horizon $P$, update rate $\eta$, iteration number $N$, previous control input $\mathbf{u}(k)$, reference $[\mathbf{y_r}(k+1)], ..., \mathbf{y_r}(k+P)]$, previous output layer bias $\mathbf{b_2}(k)$

**Output:** new output layer bias $\mathbf{b_2}(k+1)$, control input $\mathbf{u}(k+1)$

/* Initialization */

for $m = 1 : M$ do
  $\mathbf{u}(k+m) = \mathbf{u}(k)$
end

/* STEP 1: Calculate optimal control input sequence U */

for $n = 1 : N$ do
  calculate $J_{MPC}(n)$ using (4)
  calculate $\partial J_{MPC}(n)/\partial \mathbf{U}(n)$
  update $\mathbf{U}(n+1)$ using (5) and (6);
end

output optimized $\mathbf{u}(k+1)$ to the plant and the neural network, record the outputs $\mathbf{y}(k+1)$ and $\mathbf{y}_{NN}(k+1)$, respectively.

/* STEP 2: Online update the bias $\mathbf{b_2}(k+1)$ */

calculate $\mathbf{e}(k+1) = \mathbf{y}(k+1) - \mathbf{y}_{NN}(k+1)$

if $\mathbf{e}(k+1) \neq 0$ then
  $\mathbf{b_2}(k+1) = \mathbf{b_2}(k) + \mathbf{e}(k+1)$
else
  $\mathbf{b_2}(k+1) = \mathbf{b_2}(k)$
end

system output $\mathbf{y_{NN}}(k)$ at the $k$th sampling period as $\mathbf{e}\,(k) = \mathbf{y}(k) - \mathbf{y_{NN}}(k)$ and assume that the error is constant within the prediction horizon. Meanwhile, according to the MPC control process, multistep predictions need to be generated based on the prediction horizon and the available measurements from the system. Specifically, by defining $\mathbf{y_{NN}}(k+j)$ as the prediction for the system output $\mathbf{y}(k+j)$ based on the available measurements at the $k$th sampling period, the multistep prediction procedure is described as follows:

$$\begin{aligned}\mathbf{y_{NN}}\,(k+j-i) &= \mathbf{y}\,(k+j-i)\,,\ \forall i \geq j,\\ \mathbf{y_{NN}}\,(k+j) &= \mathbf{w}_2\,\sigma\,(\mathbf{w}_1\mathbf{x_{NN}}\,(k+j) + \mathbf{b}_1)\\ &\quad + \mathbf{b}_2\,(k+j)\,,\ 1 \leq j \leq P \end{aligned} \tag{7}$$

where $\mathbf{x_{NN}}(k+j) = [\mathbf{y_{NN}}(k+j-1), \ldots, \mathbf{y_{NN}}(k+j-n_y), \mathbf{u_{NN}}(k+j-1), \ldots, \mathbf{u_{NN}}(k+j-n_u)]$ is the input feature vector predicted by the NN to the MPC model.

*Proposition 1:* The nonlinear NN-based MPC system (4) has no steady-state error using the update steps in Algorithm 1 if 1) the closed-loop system is stable and 2) the reference $\mathbf{y_r}$ is reachable [22].

*Proof:* Assumption 1) implies that the closed-loop system can reach an asymptotically stable equilibrium point defined by the following:

$$\mathbf{y}^{\infty} = f\,(\mathbf{y}^{\infty}, \ldots, \mathbf{y}^{\infty}, \mathbf{u}^{\infty}, \ldots, \mathbf{u}^{\infty}) \tag{8}$$

where the superscript $\infty$ denotes the steady state, and $\mathbf{y}^{\infty}$, $\mathbf{u}^{\infty}$ are the system's steady-state output and input, respectively. From (2), the model output

$$\mathbf{y}^{\infty}_{\boldsymbol{NN}} = \mathbf{w}_2\,\sigma\,(\mathbf{w}_1\mathbf{x}^{\infty} + \mathbf{b}_1) + \mathbf{b}_2^{\infty} \tag{9}$$

is a constant at the process's steady state $\{\mathbf{y}^{\infty}, \mathbf{u}^{\infty}\}$, where $\mathbf{x}^{\infty} = [\mathbf{y}^{\infty}, \ldots, \mathbf{y}^{\infty}, \mathbf{u}^{\infty}, \ldots, \mathbf{u}^{\infty}]$. Then, the model mismatch $\mathbf{e}^{\infty} = \mathbf{y}^{\infty} - \mathbf{y}^{\infty}_{NN} \neq 0$ is always a constant within the prediction horizon. According to the update steps in Algorithm 1, we obtain the one-step-ahead prediction

$$\begin{aligned}\mathbf{y}^{\infty}_{\boldsymbol{NN}}\,(k+1) &= \mathbf{w}_2\sigma\,(\mathbf{w}_1\mathbf{x}^{\infty}\,(k+1) + \mathbf{b}_1)\\ &\quad + \mathbf{b}_2^{\infty}\,(k+1)\,.\end{aligned} \tag{10}$$

Since

$$\begin{aligned}\mathbf{x}^{\infty}\,(k+1) &= [\mathbf{y}^{\infty}_{\boldsymbol{NN}}\,(k)\,, \ldots, \mathbf{y}^{\infty}_{\boldsymbol{NN}}\,(k+1-n_y)\,, \mathbf{u}^{\infty}, \ldots, \mathbf{u}^{\infty}]\\ &= [\mathbf{y}^{\infty}, \ldots, \mathbf{y}^{\infty}, \mathbf{u}^{\infty}, \ldots, \mathbf{u}^{\infty}] = \mathbf{x}^{\infty}\,,\\ \mathbf{b}_2^{\infty}\,(k+1) &= \mathbf{b}_2^{\infty} + \mathbf{e}^{\infty}\end{aligned}$$

(10) becomes $\mathbf{y}^{\infty}_{\boldsymbol{NN}}(k+1) = \mathbf{y}^{\infty}_{\boldsymbol{NN}} + \mathbf{e}^{\infty} = \mathbf{y}^{\infty}$.

For the two-step-ahead prediction

$$\begin{aligned}\mathbf{y}^{\infty}_{\boldsymbol{NN}}\,(k+2) &= \mathbf{w}_2\,\sigma\,(\mathbf{w}_1\mathbf{x}^{\infty}\,(k+2) + \mathbf{b}_1)\\ &\quad + \mathbf{b}_2^{\infty}\,(k+2)\end{aligned} \tag{11}$$

where

$$\begin{aligned}\mathbf{x}\,(k+2) &= [\mathbf{y_{NN}}\,(k+1)\,, \mathbf{y}^{\infty} \ldots, \mathbf{y}^{\infty}, \mathbf{u}^{\infty}, \ldots, \mathbf{u}^{\infty}]\\ &= [\mathbf{y}^{\infty}, \ldots, \mathbf{y}^{\infty}, \mathbf{u}^{\infty}, \ldots, \mathbf{u}^{\infty}] = \mathbf{x}^{\infty}\,,\\ \mathbf{b}_2^{\infty}\,(k+2) &= \mathbf{b}_2^{\infty}\,(k+1) = \mathbf{b}_2^{\infty} + \mathbf{e}^{\infty}\end{aligned}$$

(11) becomes $\mathbf{y}^{\infty}_{\boldsymbol{NN}}(k+2) = \mathbf{y}^{\infty}_{\boldsymbol{NN}} + \mathbf{e}^{\infty} = \mathbf{y}^{\infty}$.

Recursively, for step-ahead prediction of three or more

$$\mathbf{y}^{\infty}_{\boldsymbol{NN}}\,(k+j) = \mathbf{y}^{\infty},\ j = 34, \ldots, P. \tag{12}$$

Meanwhile, when the process reaches a steady state, $\Delta\mathbf{u}^{\infty}\,(k+j) = \mathbf{u}^{\infty}\,(k+j) - \mathbf{u}^{\infty}\,(k+j-1) = 0$, thus allowing us to reduce the cost function (4) to

$$J_{\text{MPC}} = \sum_{j=1}^{P} \|\mathbf{y_r}\,(k+j) - \mathbf{y}^{\infty}_{\boldsymbol{NN}}\,(k+j)\,\|^2. \tag{13}$$

Since $\mathbf{y}^{\infty}_{\boldsymbol{NN}}(k+j) = \mathbf{y}^{\infty}, j = 1, 2, \ldots, P$, for the ideal solution and considering assumption (b), we can conclude that $\mathbf{y_r} = \mathbf{y}^{\infty}$ and there is no steady-state error.

### D. Stability Analysis

In *Proposition 1*, we have assumed that the closed-loop system is stable. In fact, the stability of the closed loop is of utmost importance for any application of nonlinear MPC to a real process. Many contributions have been made to tackle the stability problems in a state-space framework. These methods either use a cost function as a Lyapunov candidate function or require some state to decrease in a certain norm. However, in the proposed method, the system is modeled by a neural network model rather than a state-space mode meaning the stability analysis based on state-space cannot be applied. Therefore, we propose a method to achieve stability by imposing a terminal constraint on outputs to ensure the monotonicity of the cost function as established in [23], [24]. This requires the addition of a terminal constraint to (4) to ensure stability. The terminal constraint is defined as follows:

$$y_r\,(k+P+j) - y_{NN}\,(k+P+j) = 0,\ \forall j \in [1, N_c] \tag{14}$$

where $N_c$ is the constraint horizon.

*Proposition 2:* The nonlinear MPC controller is asymptotically stable if the following conditions are satisfied:

1) $\lambda \neq 0$;
2) $N_c = \max[n_y + 1,\ n_u + 1 + M - P]$ .

*Proof:* Consider the cost function

$$J_{\text{MPC}}\ (k) = \sum_{j=1}^{P} \|\mathrm{e}\,(k+j)\,\|^2\ + \lambda \sum_{j\,=\,1}^{M} \|\Delta \mathrm{u}\,(k+j)\,\|^2 \quad (15)$$

at time $k$ , where $\mathrm{e}\,(k+j) = \mathrm{y}_r\,(k+j) - \mathrm{y}_{NN}(k+j)$. Let us assume that $\mathbf{U}\,(k) = [\mathbf{u}(k+1),\ \mathbf{u}(k+2), \ldots, \mathbf{u}(k+M)]$ is the optimal control at time $k$ found by an optimization procedure. Suppose the suboptimal control $\mathbf{U}^*(k+1)$ postulated at time $k+1$ is

$$\mathbf{U}^*\,(k{+}1) = [\mathbf{u}\,(k{+}2)\,, \mathbf{u}\,(k{+}3)\,, \ldots, \mathbf{u}\,(k{+}M)\,,\ \mathbf{u}\,(k{+}M)]\,. \quad (16)$$

The control sequence $\mathbf{U}^*(k+1)$ is formed based on the control derived at time $k$. Assume that the model mismatches and unmeasured disturbances are constant within the prediction horizon, and the predictions $\mathrm{y}_{NN}(k+j)$ derived at time $k+1$ are the same as those derived at time $k$. For the suboptimal control $\mathbf{U}^*(k+1)$, the cost function can be defined as follows:

$$J_{\text{MPC}}{}^*\ (k+1) = \sum_{j=2}^{P+1} \|\mathrm{e}\,(k+j)\,\|^2\ + \lambda \sum_{j\,=\,2}^{M} \|\Delta \mathrm{u}\,(k+j)\,\|^2. \quad (17)$$

The difference of cost functions $J_{\text{MPC}}(k)$ and $J_{\text{MPC}}{}^*(k+1)$ can be calculated as follows:

$$J_{\text{MPC}}{}^*\,(k{+}1) - J_{\text{MPC}}\ (k) = \|\mathrm{e}\,(k{+}P{+}1)\,\|^2\ - \|\mathrm{e}\,(k+1)\,\|^2 - \lambda \|\Delta \mathrm{u}\,(k+1)\,\|^2. \quad (18)$$

From the terminal equality constraints (14), $\|\mathrm{e}(k+P+1)\|^2 = 0$, we obtain the following:

$$J_{\text{MPC}}{}^*\,(k{+}1) - J_{\text{MPC}}\ (k) = -\|\mathrm{e}\,(k{+}1)\,\|^2 - \lambda \|\Delta \mathrm{u}\,(k+1)\,\|^2. \quad (19)$$

The terminal equality constraints hold for all $j \geq 1$ if
1) $N_c \geq n_y + 1$ , if $n_y \geq n_u + 1 + M - P$;
2) $N_c \geq n_u + 1 + M - P$ , if $n_y < n_u + 1 + M - P$.

These two conditions lead to the constraint horizon

$$N_c = \max\,[n_y + 1,\ n_u + 1 + M - P] \quad (20)$$

which guarantees that the equality constraints of tracking error hold for all $j \geq 1$. From the definition of $\mathbf{U}^*(k+1)$ in (16), we can infer that constraint (14) is satisfied at time $k+1$. Furthermore, if $\mathbf{U}(k+1)$ is the optimal solution to the optimization problem at time $k+1$, then $J_{\text{MPC}}(k+1) \leq J_{\text{MPC}}{}^*(k+1)$. Finally

$$\Delta J_{\text{MPC}}\,(k+1) \leq -\|\mathrm{e}\,(k+1)\,\|^2 - \lambda \|\Delta \mathrm{u}\,(k+1)\,\|^2. \quad (21)$$

For $\lambda \neq 0$, it is clear to see that the cost function monotonically decreases with respect to time and the control system is stable. The constrained MPC optimization problem can be solved according to the algorithm presented in [23].

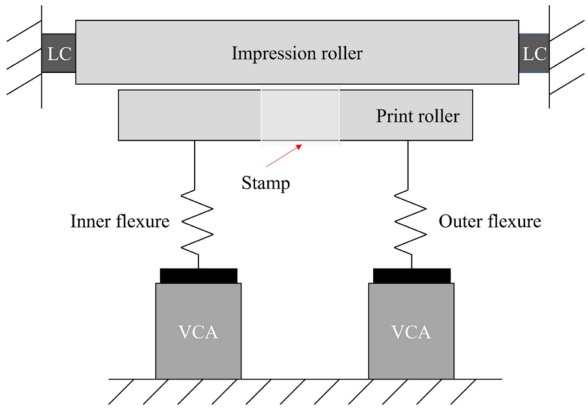


Fig. 2. Schematic diagram of the flexure based R2R contact printing system.

## III. Experiments and Discussions

In this section, we will demonstrate an R2R microcontact printing prototype where we test the mN-scale contact force control using our NN-based MPC. We generate the training dataset for system identification and NN architecture initialization using a pulse-train method in Section III-B. In Section III-C, we first use simulations to tune the parameters for the MPC. Then, we measure the performance of the contact force control broadly using various reference signals, pressure-sensitive films, and straight-line pattern printings.

### A. Experimental Setup

The proposed control method was evaluated on a flexure-based R2R contact printing system as shown in Fig. 1. It consists of two subsystems: the web handling system and the printing module. The web handling system is composed of two motorized rollers and two idler rollers. A ring rotation encoder and a readhead were mounted on the left idler roller to measure the linear web moving speed. The measured speed is then used as a feedback signal to control the web moving speed. The right idler roller is a tension measuring roller, which measures the tension of the moving web for the web tension control. As the web tension and motion speed are correlated in the R2R system [25], [26], the web tension and the linear speed of the moving web need to be synchronously controlled. In Fig. 1, the motorized unwind roller is controlled to maintain a constant linear web moving speed using the feedback signal from the speed encoder; the motorized rewind roller is operated under torque control mode to maintain a constant web tension using the feedback signal from the tension measuring roller.

Fig. 2 shows the schematic diagram of the flexure-based R2R contact printing system, where the springs at the two ends of the print roller represent the inner flexure and the outer flexure. Before printing, a stamp must be wrapped onto the print roller. During printing, the inked stamp is actuated by a positioning stage to transfer the patterns on the stamp to the flexible web substrate by compressing the inked stamp against the web substrate. The positioning stage is made up of two parallel plate

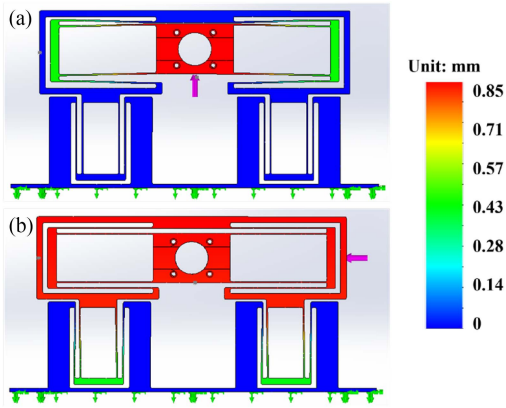


Fig. 3. Simulated motions of a single flexure mechanism. (a) Vertical direction. (b) Horizontal direction.

flexures with each one connected to the print roller shaft via air bushings. Each flexure has two degrees of freedom with a $\pm 3$ mm range of motion (ROM) in the vertical axis, and a $\pm 1$ mm ROM in the horizontal direction as simulated in Fig. 3. The two ends of the impression roller are clamped to two load cells (MAGPOWR TS25FC, MAXCESS) to measure the contact force on both ends with a resolution of 0.03 N. The print roller is driven by two voice coil actuators (VCAs) (NCC08-34-350-2X, H2W TECHNOLOGIES) through the flexure mechanism. The VCAs are connected to linear current amplifiers (LCA-5/15, H2W TECHNOLOGIES) that generate the driving currents linearly proportional to the command voltages. The VCAs provide force to push the motion stage and the print roller in the vertical direction to bring the stamp in contact with the web. The command voltages are from the real-time controller (NI CompactRIO 9049) where all the sensors and amplifiers are integrated. The amplifier and VCA constants are 0.1 A/V and 59.6 N/A. The control loop rate is set at 10 ms. In summary, the control inputs are the voltage commands from the real-time controller and the controlled variables [refer to $\mathbf{y}$ in (1)] are the contact force measured by the two load cells. The main objective of the contact force control is to maintain a constant force at the contact areas between the stamp on the print roller and substrate surface. As the print roller is force-controlled at both ends, it also has an alternative objective of balancing the force at the two ends for large-area printing.

The positioning system in this microcontact printing module suffers various disturbances from both the web handling system and the positioning system itself, including misalignment, cylindricity, and fabrication errors. These disturbances produce nonlinear effects that are difficult to manage with conventional control systems as shown in Fig. 4. We will demonstrate the excellent performance of the proposed NN-based adaptive MPC method for precise contact force control on this module in the following sections.

### B. Model Identification

To capture the dynamics of the flexure mechanism, we used a pulse-train method to generate the training dataset. Figs. 5 and 6 show the series of pulses with random amplitudes and the responses of the plant. Each pulse lasts one second so that the plant can settle to a stable state during the pulse signal. The sampling time is set to 10 ms which is the same as the control loop rate. To compensate for the imbalance of the flexure mechanism, which may be caused by the misalignment in the system, fault in calibration, or both, the input voltages to the inner VCAs and the outer VCAs are randomly selected in [4 V, 5.5 V] and [0.5 V, 1.5 V], respectively. The corresponding forces measured from load cells are in the range of 8–18 N as shown in Fig. 6. To sufficiently span the working space of the plant, 50000 samples are generated. In this experiment, we use 80% of the data for training, and 20% for testing.

Since the contact force is affected by both its output value history and the input control commands, the dynamics of the plant can be approximated by the NARX function in (1). The time delay for the input and the output is $n_u = n_y = 2$ and

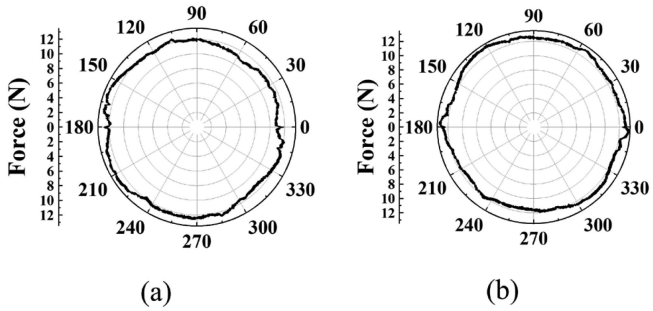


Fig. 4. Open-loop performance of the flexure mechanism. (a) Readings from the inner load cell. (b) Readings from the outer load cell.

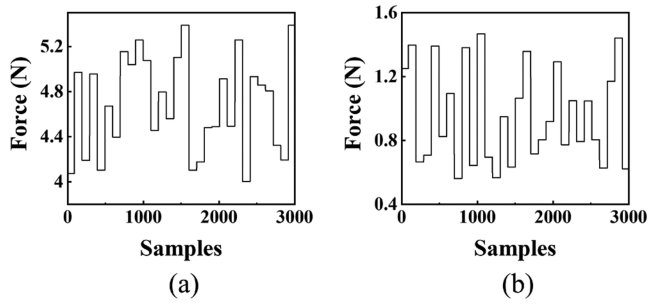


Fig. 5. Pulse train examples used as the input to the plant. (a) Inputs to the inner VCA. (b) Inputs to the outer VCA.

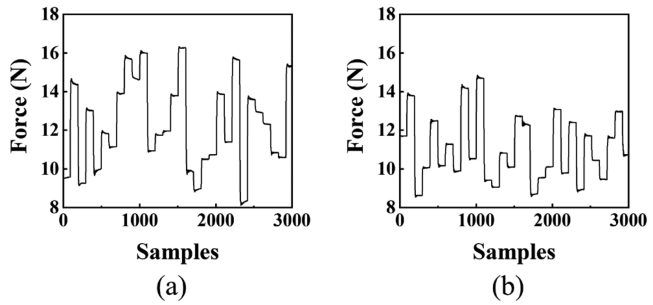


Fig. 6. Responses of the plant due to the pulse train inputs. (a) Responses of the inner flexure. (b) Responses of the outer flexure.

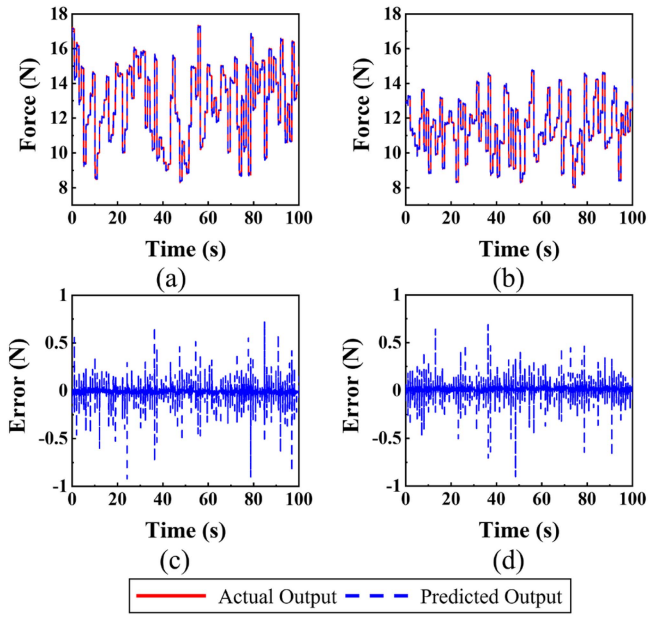


Fig. 7. Actual outputs and predicted outputs of the flexure mechanism of (a) inner flexure and (b) outer flexure, and the corresponding errors (c), (d), respectively.

TABLE I
COMPARISON RESULTS OF DIFFERENT PREDICTIVE MODELS

| Models | RMSE (N) | Running time (s) |
|---|---|---|
| GP | 0.077 | 304.153 |
| GMM | 0.057 | 0.020 |
| MSVR | 0.072 | 2.048 |
| NN | **0.048** | **0.015** |

Note: The winner is shown in boldface.

the number of hidden nodes in the NN is chosen to be 7. With these values in mind, the dimensions of $\mathbf{w}_1$, $\mathbf{b}_1$, $\mathbf{w}_2$, and $\mathbf{b}_2$ in (2), respectively, become $7 \times 8$ , $7 \times 1$, $2 \times 7$, and $2 \times 1$. The tangent-sigmoid function is chosen as the activation function since the training data are normalized between [–1], [1]. The NN is trained using the Levenberg–Marquardt algorithm in the MATLAB R2019a environment. Fig. 7 shows the prediction results for the testing data, with the RMSE of 0.048 N. In order to examine the predictive accuracy and inference speed of the proposed method, the NN model is compared with a Gaussian process model (GP) [27], a Gaussian mixture model (GMM) [28], and a multivariate support vector regression model (MSVR) [29]. From the testing results listed in Table I, the NN model shows superior accuracy and prediction speed.

### C. Simulations and Predictive Control

First, to evaluate the proposed method, simulations are carried out to tune the parameters for the MPC, including the control horizon, the prediction horizon, and the weighting factor $\lambda$. To tune the control horizon and the prediction horizon, we summarize the average RMSE of contact force for three different signals, namely step signals, ramp signals, and sinusoidal signals shown in Table II by varying $M$ from 1 to 10 intervals and $P$ from 1 to 20 intervals. Note that Table II is upper triangular because $M$ cannot be larger than $P$. When $P > 7$ and $M > 2$ sampling intervals, the improvement in the system performance is trivial as shown in Table II. Therefore, $M = 2$ and $P = 7$ are chosen for the final control horizon and prediction horizon to achieve desirable real-time force control. For the weighting factor $\lambda$, since the system outputs and the inputs are both scaled to [–1], [1], it is reasonable to assume that $\lambda$ is greater than 0 and smaller than 1. To quantitively evaluate the effects on the control performance from different weighting factor $\lambda$, we use overshoot (%) and rise time (ms) for performance evaluation criteria. Table III shows the overshoot and rise time for different values of $\lambda$. It can be found that there is a tradeoff between the rise time and the overshoot. For example, for $\lambda = 0.1$, the rise time is shorter than that of $\lambda = 0.01$, but the overshoot is greater. We ultimately select $\lambda = 0.1$ for a balance between the rise time and the overshoot. Using the above-tuned parameters, the execution of the proposed control algorithm takes 4 ms for real-time contact force control, which is less than the control loop rate of 10 ms as mentioned in Section III–A. To demonstrate the computational performance that one might expect to get with this algorithm and the optimization, timing data was collected on the real-time controller using different values of $M$ and $P$ which are displayed in the Appendix.

Second, to test the proposed adaptive algorithm of the NN model in eliminating the steady-state error, simulations are carried out considering different updating laws and the results are shown in Fig. 8. For standard MPC without any adaptive mechanism, there will be a steady-state error as shown in Fig. 8(a). When weights and biases update together in one control loop [18], the system becomes unstable as shown in Fig. 8(b). The reason is that if weights are updated according to one sample acquired from one control loop, the NN model tends to forget the past sequential training information. Thus, the prediction error may increase for the subsequent new samples leading the controller to make improper decisions. As shown in Fig. 8(c), our proposed adaptive MPC scheme can significantly compensate the induced disturbance.

Third, contact force control experiments are carried out to test the proposed NN-based adaptive MPC controller. The performance of the proposed control scheme has been assessed using three different signals, namely step signals, ramp signals, and sinusoidal signals. The proposed control method is compared with a PID controller, an NN-based standard MPC, and an NN-based robust MPC controller [30]. Figs. 9–11 show control results for different reference signals. For clarity of presentation only, results for the proposed NN-based adaptive MPC controller and the NN-based standard MPC controller are presented. Table IV shows the control quality of different controllers, quantified by RMSE between the reference outputs and the actual outputs shown in Figs. 9–11 under different reference signals. Overall, our proposed NN-based adaptive MPC controller outperformed all other controllers.

Fourth, to further verify the robustness of our proposed control method in real printing processes, we measured and recorded the contact forces on both ends of the impression roller for one revolution (360°). Fig. 12 shows the contact force data with a

TABLE II
AVERAGE RMSE (N) OF CONTACT FORCE WHERE $M$ AND $P$ VARY BETWEEN 1–10 AND 1–20, RESPECTIVELY

| M \ P | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | - | 1.12 | 0.96 | 0.96 | 0.99 | 1.01 | 1.03 | 1.05 | 1.07 | 1.09 | 1.11 | 1.16 | 1.33 | - | - | - | - | - | - | - |
| 2 | | 1.18 | 0.94 | 0.89 | 0.87 | 0.86 | 0.85 | 0.85 | 0.85 | 0.85 | 0.85 | 0.85 | 0.84 | 0.84 | 0.85 | 1.04 | - | - | - | - |
| 3 | | | 0.95 | 0.90 | 0.88 | 0.87 | 0.86 | 0.86 | 0.86 | 0.85 | 0.85 | 0.85 | 0.85 | 0.85 | 0.85 | 0.85 | 0.88 | - | - | - |
| 4 | | | | 0.90 | 0.88 | 0.87 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.89 | 0.90 |
| 5 | | | | | 0.88 | 0.87 | 0.86 | 0.86 | 0.86 | 0.88 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.88 | 0.86 | 0.86 | 0.87 | 0.87 |
| 6 | | | | | | 0.87 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 |
| 7 | | | | | | | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.87 | 0.86 | 0.86 | 0.86 | 0.86 | 0.88 | 0.86 |
| 8 | | | | | | | | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.87 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 |
| 9 | | | | | | | | | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 |
| 10 | | | | | | | | | | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 | 0.86 |

Note: '-' indicates the system is not stable.

TABLE III
OVERSHOOT AND RISING TIME FOR DIFFERENT λ

| λ | 0.01 | 0.1 | 1 |
|---|---|---|---|
| Overshoot | 0.32% | 0.26% | 0.32% |
| Rise time | 40 ms | 50 ms | 65 ms |

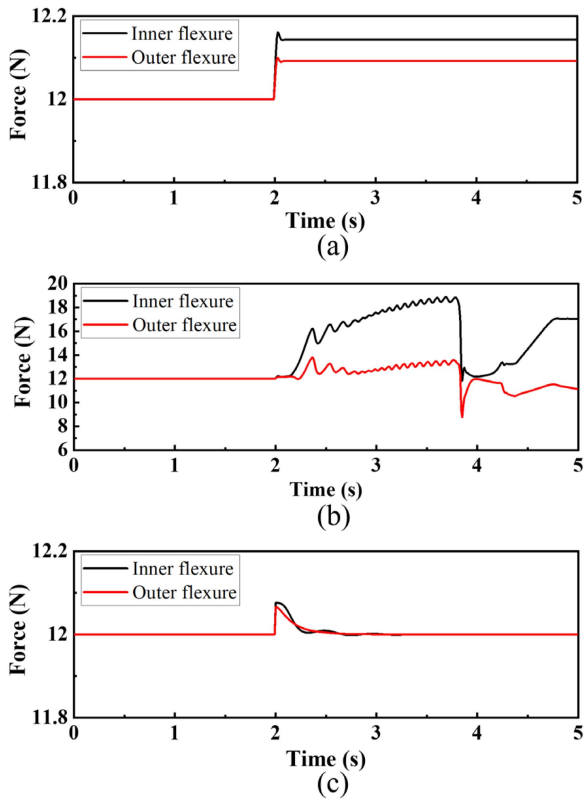


Fig. 8. Simulation results for different update laws for inner flexure and outer flexure. (a) No updating laws. (b) Updating both weights and biases. (c) Updating only biases.

reference contact force of 12 N and a rolling speed of 1 mm/s. The RMSE of the contact forces is maintained in the range of 0–0.075 N. In other words, our NN-based adaptive MPC method has achieved a desirable stable contact force control and can compensate for model mismatches and external disturbances.

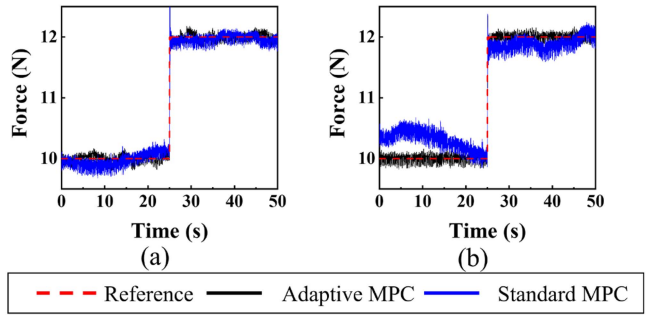


Fig. 9. Experimental results, comparison of the reference tracking results between the proposed NN-based adaptive MPC controller and the NN-based standard MPC controller of the flexure mechanism. (a) Inner flexure. (b) Outer flexure.

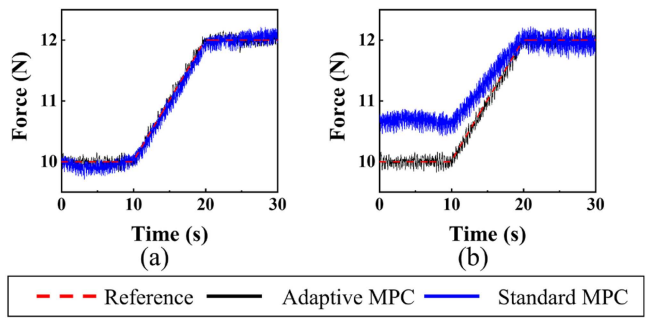


Fig. 10. Experimental results, comparison of the reference tracking results between the proposed NN-based adaptive MPC controller and the NN-based standard MPC controller of the flexure mechanism. (a) Inner flexure. (b) Outer flexure.

Moreover, we used pressure-sensitive films to mimic the web substrate in order to measure the real contact pressure from our proposed control scheme. The pressure-sensitive films change color from white to red over a certain pressure range: the higher the pressure, the more intense the color change. The pressure-sensitive films (177.8 mm × 88.9 mm, Prescale LLLLW, Fujifilm) used in our experiments have a pressure sensing range of 0.05–0.2 MPa. Fig. 13(a) and (b), respectively, show the films that were pressed by the print roller using the NN-based standard MPC controller and the proposed NN-based adaptive MPC controller. Our proposed MPC controller resulted in a visually more uniform color (pressure) distribution on the

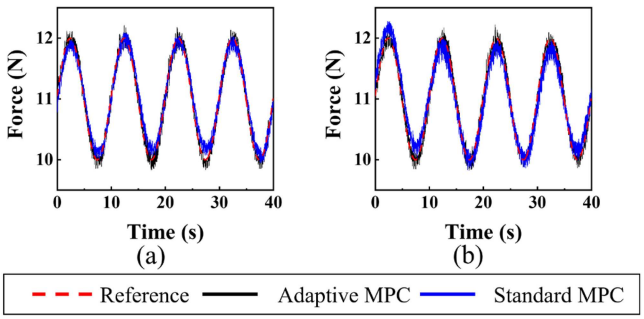


Fig. 11. Experimental results, comparison of the reference tracking results between the proposed NN-based adaptive MPC controller and the NN-based standard MPC controller of the flexure mechanism. (a) Inner flexure. (b) Outer flexure.

TABLE IV
CONTROL RESULTS (RMSE) FOR DIFFERENT REFERENCE SIGNALS

| | Step | Ramp | Sinusoidal |
|---|---|---|---|
| PID | 0.0918 N | 0.0969 N | 0.0963 N |
| NN-based standard MPC | 0.2099 N | 0.3303 N | 0.1551 N |
| NN-based robust MPC | 0.1206 N | 0.1324 N | 0.1133 N |
| NN-based adaptive MPC | **0.0761 N** | **0.0647 N** | **0.0950 N** |

Note: The winner is shown in boldface.

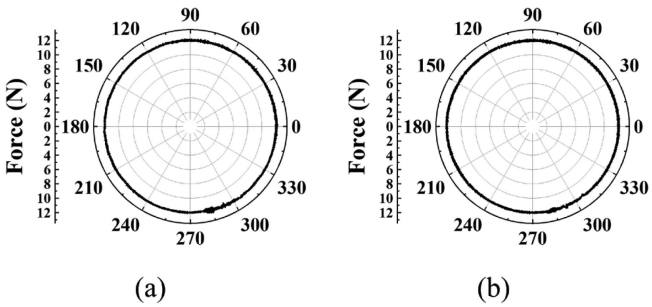


Fig. 12. Measured printing forces for one revolution when the printing speed is 1 mm/s. (a) Inner flexure. (b) Outer flexure.

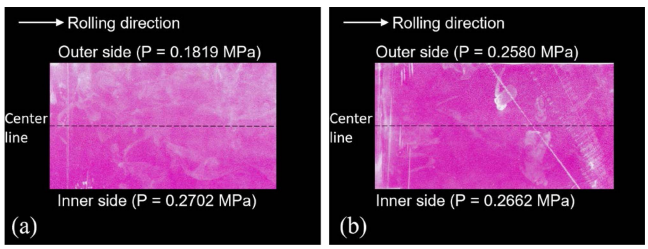


Fig. 13. Pressure-sensitive film test results. (a) NN-based standard MPC controller. (b) Proposed NN-based adaptive MPC controller.

film compared to the alternative MPC control. To quantify the pressure values obtained on the films, we have evaluated the results by mapping the color density onto the amount of applied pressure. First, the films were divided into two regions evenly along the web rolling direction, namely the inner portion and the outer portion. Then, we calculated the average color densities in each region. Lastly, we mapped the color densities to the

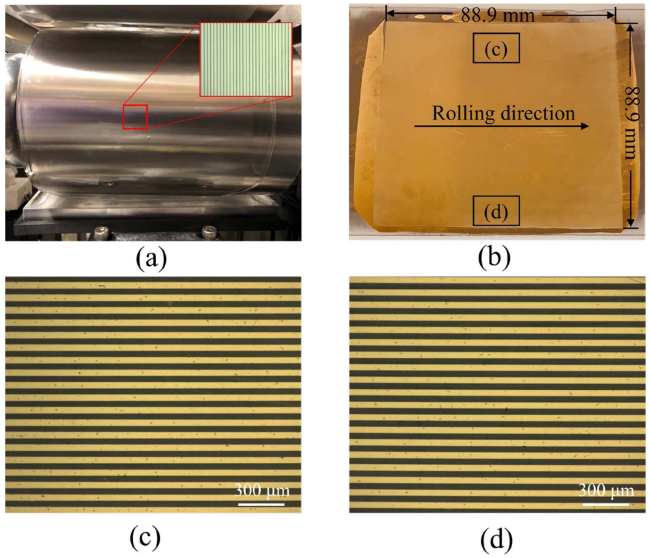


Fig. 14. Large-area gold patterns printed on flexible PET substrates based on the proposed NN-based MPC controller. (a) PDMS stamp with line features wrapped around the print roller. (b) Photo of printed large-area gold patterns on flexible PET substates. The size of the patterned area is 88.9 mm × 88.9 mm. (c), (d) Magnified images of printed patterns on the inner side and the outer side respectively, which show the uniform printing results achieved by the proposed NN-based adaptive MPC controller.

TABLE V
MEAN AND STANDARD DEVIATION OF THE GOLD LINE WIDTH MEASURED FROM THE INNER PORTION AND THE OUTER PORTION OF THE ETCHED SAMPLE

| | Mean (μm) | Standard deviation (μm) |
|---|---|---|
| Inner | 45.02 | 0.50 |
| Outer | 45.31 | 0.50 |

amounts of applied pressure using the colorimetric calibration curve that we obtained experimentally. The quantified pressure values are shown in Fig. 13. The difference of applied pressure on the inner and outer side of the film from the proposed NN-based adaptive MPC controller is 0.0082 MPa, 10 times smaller than the difference of the NN-based standard MPC, whose difference is 0.0883 MPa. Please note, the observable defects on the pressure films originated from the raw film material but not from the contact force control.

Finally, line patterns of 45 $\mu$m width and 55 $\mu$m pitch were printed using our R2R microcontact printer to show the effectiveness of the proposed control method. We follow the standard R2R microcontact printing process, which include four steps: stamp preparation; inking of the stamp; R2R printing; and etching. We set the rolling speed at 1 mm/s, and the contact force at 15 N. Fig. 14(a) shows the inked polydimethylsiloxane (PDMS) stamp wrapped around the print roller. Fig. 14(b) shows the printed gold patterns on flexible polyethylene terephthalate (PET) substrates. The patterned area is 88.9 mm × 88.9 mm. Fig. 14(c) and (d) are the magnified images of the printed line patterns on the inner and outer side of the substrate. Fig. 14 shows highly uniform printing results from the proposed NN-based

TABLE VI
COMPUTATION TIME (ms) FOR THE PROPOSED METHOD WHERE $M$ AND $P$ VARY BETWEEN 1–10 AND 1–20, RESPECTIVELY

| M \ P | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 | 11 | 12 | 13 | 14 | 15 | 16 | 17 | 18 | 19 | 20 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | 0.56 | 1.07 | 1.59 | 2.1 | 2.62 | 3.13 | 3.65 | 4.17 | 4.7 | 5.2 | 5.71 | 6.23 | 6.75 | 7.26 | 7.78 | 8.3 | 8.8 | 9.3 | 9.84 | 10.35 |
| 2 | | 1.3 | 1.9 | 2.47 | 3.05 | 3.63 | 4.22 | 4.8 | 5.4 | 6 | 6.55 | 7.14 | 7.72 | 8.32 | 8.9 | 9.48 | 10.05 | 10.65 | 11.23 | 11.8 |
| 3 | | | 2 | 2.61 | 3.22 | 3.83 | 4.44 | 5.05 | 5.68 | 6.28 | 6.9 | 7.5 | 8.1 | 8.72 | 9.34 | 9.93 | 10.55 | 11.2 | 11.81 | 12.4 |
| 4 | | | | 2.74 | 3.4 | 4.02 | 4.65 | 5.31 | 5.94 | 6.58 | 7.21 | 7.85 | 8.48 | 9.12 | 9.8 | 10.39 | 11.04 | 11.7 | 12.32 | 12.93 |
| 5 | | | | | 3.53 | 4.2 | 4.88 | 5.55 | 6.2 | 6.86 | 7.55 | 8.21 | 8.86 | 9.54 | 10.2 | 10.85 | 11.51 | 12.18 | 12.84 | 13.53 |
| 6 | | | | | | 4.38 | 5.05 | 5.75 | 6.45 | 7.15 | 7.85 | 8.52 | 9.22 | 9.93 | 10.63 | 11.29 | 12 | 12.69 | 13.4 | 14.1 |
| 7 | | | | | | | 5.25 | 6 | 6.7 | 7.41 | 8.13 | 8.85 | 9.6 | 10.29 | 11.03 | 11.72 | 12.5 | 13.2 | 13.9 | 14.65 |
| 8 | | | | | | | | 6.18 | 6.93 | 7.68 | 8.44 | 9.2 | 9.92 | 10.68 | 11.45 | 12.19 | 12.93 | 13.7 | 14.43 | 15.2 |
| 9 | | | | | | | | | 7.18 | 7.95 | 8.73 | 9.49 | 10.28 | 11.04 | 11.84 | 12.61 | 13.43 | 14.2 | 15.03 | 15.75 |
| 10 | | | | | | | | | | 8.22 | 9.04 | 9.83 | 10.65 | 11.44 | 12.25 | 13.1 | 13.9 | 14.7 | 15.53 | 16.3 |

TABLE VII
PERCENTAGE OF TIME FOR KEY ROUTINES WHERE $M = 2$ AND $P = 7$

| Routine | Percent time |
|---|---|
| Gradient calculation | 96 |
| prediction | 4 |
| Misc. | <1 |

adaptive MPC controller and demonstrates that the proposed control algorithm can be used for large-area R2R contact printing. In addition to the qualitative visualization of the printing results, quantitative quality measurements were performed on the etched sample. We used a Mitutoyo Quick Vision ACTIVE 202 (Mitutoyo American Corporation, Marlborough MA, USA) with an object space resolution of 1.04 $\mu$m to measure the mean and standard deviation of the gold line width. The measured locations were selected to be the inner and outer portions of the etched sample as shown in Fig. 14(c) and (d) to illustrate the spatial variation in print quality. Table V shows the measured results. To summarize, the difference of the average gold line width on the inner and outer portions is approximately 0.3 $\mu$m, which is smaller than the object space resolution of the imaging system. Additionally, both standard deviations are 0.5 $\mu$m, respectively. With this knowledge, we can conclude that the printing quality, in every practical sense, is the same for the inner and outer portions of the etched sample.

## IV. CONCLUSION

In this article, we have developed an NN-based adaptive MPC controller for a flexure-based R2R contact printing system. The nonlinearity of the flexure mechanism is learned by an NN. To eliminate the steady-state error in real experiments, an NN-based adaptive MPC controller is designed, where the biases of the output layer of the NN model are updated during every control loop according to the prediction error between the NN output and the actual plant output. Experimental results show that the RMSE of the contact force is maintained at values less than 0. 075 N with balances on both ends of the impression roller for rolling speeds of 1 mm/s. Uniform microscale gold pattern printing results have shown the effectiveness of the proposed NN-based adaptive MPC controller in the actual printing process.

Our proposed NN-based MPC can be easily applied for control of other nonlinear manufacturing processes. The method even shows potential for other complex control problems in the realm of roll-to-roll manufacturing systems to reduce trial and error commonly in physical model based control, such as the web tension control [31] and the register control [32], [33]. Specifically, the complex models of web tension disturbances and the register errors can be learned for control using our proposed NN-based adaptive MPC for the web tension control and the register control.

The limitation of the proposed method is that only short-term temporal information is considered. In fact, for time-dependent tasks with sequential inputs, both long-term and short-term information affects the response of the system. To integrate long-term information from the system into the predictive model, long short-term memory NN structure can be further studied to improve the modeling accuracy.

## APPENDIX

To demonstrate the computational performance that one might expect to get with this algorithm and the optimization, timing data were collected on the real-time controller for different values of $M$ and $P$. The processor of the real-time controller (NI CompactRIO 9049) is a 1.60 GHz Quad-Core CPU (Intel Atom E3940). The architecture of the neural network model was the same as the model used in Section III. The data shown in Table VI represent the computation time for the proposed method where the number of iterations taken to produce a solution was set to 100 as described in Section III. The values for $M$ ranged from 1 to 10 and the values for $P$ varied from 1 to 20. Note that the Table VI is upper triangular because $M$ cannot be larger than $P$. The unit is in milliseconds. The computational cost of the proposed method can be broken down into 3 separate costs. The gradient calculation, the plant prediction, and other miscellaneous operations are calculated based on a percentage of computational cost. The case where $M = 2$ and $P = 7$ is presented in Table VII. From Table VII, one can find that the gradient calculation occupies most of the computation time.

## REFERENCES

[1] K. Hwang *et al.*, "Toward large scale Roll-to-Roll production of fully printed perovskite solar cells," *Adv. Mater.*, vol. 27, no. 7, pp. 1241–1247, 2015.
[2] C. Y. Hui, A. Jagota, Y. Y. Lin, and E. J. Kramer, "Constraints on microcontact printing imposed by stamp deformation," *Langmuir*, vol. 18, no. 4, pp. 1394–1407, Feb. 2002.

[3] F. Lazzarino, C. Gourgon, P. Schiavone, and C. Perret, "Mold deformation in nanoimprint lithography," *J. Vac. Sci. Technol. B, Microelectron. Nanometer Struct. Process., Meas., Phenomena*, vol. 22, no. 6, pp. 3318–3322, Nov. 2004.
[4] T.-M. Lee, J.-H. Noh, I. Kim, D.-S. Kim, and S. Chun, "Reliability of gravure offset printing under various printing conditions," *J. Appl. Phys.*, vol. 108, no. 10, Nov. 2010, Art. no. 102802.
[5] J. E. Petrzelka and D. E. Hardt, "Static load-displacement behavior of PDMS microfeatures for soft lithography," *J. Micromech. Microeng.*, vol. 22, no. 7, Jun. 2012, Art. no. 075015.
[6] F. Wang *et al.*, "A novel actuator-internal micro/nano positioning stage with an arch-shape bridge-type amplifier," *IEEE Trans. Ind. Electron.*, vol. 66, no. 12, pp. 9161–9172, Dec. 2019.
[7] Y.-M. Choi, D. Kang, S. Lim, M. G. Lee, and S.-H. Lee, "High-Precision printing force control system for Roll-to-Roll manufacturing," *IEEE/ASME Trans. Mechatronics*, vol. 22, no. 5, pp. 2351–2358, Oct. 2017.
[8] M. M. Amjad, "Modeling of electrical grid systems to evaluate sustainable electricity generation in Pakistan," master's theses, Univ. Massachusetts Amherst, Amherst, MA, USA, Jul. 2020.
[9] C. Li, H. Xu, and S.-C. Chen, "Design of a precision multi-layer roll-to-roll printing system," *Precis. Eng.*, vol. 66, pp. 564–576, Nov. 2020.
[10] X. Zhou, D. Wang, J. Wang, and S.-C. Chen, "Precision design and control of a flexure-based roll-to-roll printing system," *Precis. Eng.*, vol. 45, pp. 332–341, Jul. 2016.
[11] J. Liu, S. Kang, W. Chen, W. Chen, and J. Jiang, "A novel flexure-based register system for R2R electronic printing," *Microsyst. Technol.*, vol. 21, no. 11, pp. 2347–2358, Nov. 2015.
[12] U. Rosolia, X. Zhang, and F. Borrelli, "Data-driven predictive control for autonomous systems," *Annu. Rev. Control, Robot., Auton. Syst.*, vol. 1, no. 1, pp. 259–286, 2018.
[13] S. Xie and J. Ren, "Recurrent-neural-network-based predictive control of piezo actuators for trajectory tracking," *IEEE/ASME Trans. Mechatronics*, vol. 24, no. 6, pp. 2885–2896, Dec. 2019.
[14] D. Knittel, E. Laroche, D. Gigan, and H. Koc, "Tension control for winding systems with two-degrees-of-freedom H/sub /spl infin//controllers," *IEEE Trans. Ind. Appl.*, vol. 39, no. 1, pp. 113–120, Jan. 2003.
[15] K. H. Shin, "Distributed control of tension in multi-span web transport systems," Ph.D. dissertation, Okalahoma Univ., Stillwater, OK, USA, 1991.
[16] H. Wang and J. Wu, "Eliminating the DC component in steady state tracking error for unknown nonlinear systems: A combination of fuzzy logic and a PI outer loop," in *Proc. Amer. Control Conf., ACC*, 1998, vol. 3, pp. 1415–1416.
[17] A. Fatehi, H. Sadjadian, A. Khaki-Sedigh, and A. Jazayeri, "Disturbance rejection in neural network model predictive control," *IFAC Proc. Vol.*, vol. 41, no. 2, pp. 3527–3532, Jan. 2008.
[18] R. Hedjar, "Adaptive neural network model predictive control," *Int. J. Innov. Comput., Inf. Control*, vol. 9, no. 3, pp. 1245–1257, 2013.
[19] B. M. Åkesson, H. T. Toivonen, J. B. Waller, and R. H. Nyström, "Neural network approximation of a nonlinear model predictive controller applied to a pH neutralization process," *Comput. Chem. Eng.*, vol. 29, no. 2, pp. 323–335, Jan. 2005.
[20] T. Chen and H. Chen, "Universal approximation to nonlinear operators by neural networks with arbitrary activation functions and its application to dynamical systems," *IEEE Trans. Neural Netw.*, vol. 6, no. 4, pp. 911–917, Jul. 1995.
[21] D. Q. Mayne, J. B. Rawlings, C. V. Rao, and P. O. M. Scokaert, "Constrained model predictive control: Stability and optimality," *Automatica*, vol. 36, no. 6, pp. 789–814, Jun. 2000.
[22] X. Tian, P. Wang, D. Huang, and S. Chen, "Offset-free multistep nonlinear model predictive control under plant–model mismatch," *Int. J. Adaptive Control Signal Process.*, vol. 28, nos. 3–5, pp. 444–463, 2014.
[23] K. Patan, "Neural network-based model predictive control: Fault tolerance and stability," *IEEE Trans. Control Syst. Technol.*, vol. 23, no. 3, pp. 1147–1155, May 2015.
[24] H.-G. Han, L. Zhang, Y. Hou, and J.-F. Qiao, "Nonlinear model predictive control based on a self-organizing recurrent neural network," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 2, pp. 402–415, Feb. 2016.
[25] J. Yan and X. Du, "Web tension and speed control in Roll-to-Roll systems," *Control Theory in Engineering*. London, U.K.: IntechOpen, 2020. [Online]. Available: https://www.intechopen.com/chapters/68827
[26] J. Yan and X. Du, "Real-time web tension prediction using web moving speed and natural vibration frequency," *Meas. Sci. Technol.*, vol. 31, no. 11, Sep. 2020, Art. no. 115205.
[27] M. P. Deisenroth and C. E. Rasmussen, "PILCO: A model-based and data-efficient approach to policy search," in *Proc. 28th Int. Conf. Int. Conf. Mach. Learn.*, 2011, pp. 465–472.
[28] S. Levine and P. Abbeel, "Learning neural network policies with guided policy search under unknown dynamics," in *Proc. Int. Conf. Neural Inf. Process. Syst.*, 2014, pp. 1071–1079.
[29] Y. Zhou, F.-J. Chang, L.-C. Chang, I.-F. Kao, Y.-S. Wang, and C.-C. Kang, "Multi-output support vector machine for regional multi-step-ahead PM2.5 forecasting," *Sci. Total Environ.*, vol. 651, pp. 230–240, Feb. 2019.
[30] K. Patan and P. Witczak, "Robust model predictive control using neural networks," in *Proc. IEEE Int. Symp. Intell. Control*, 2014, pp. 1107–1112.
[31] J. Lee, M. Jo, and C. Lee, "Advanced tension model for highly integrated flexible electronics in Roll-to-Roll manufacturing," *IEEE/ASME Trans. Mechatronics*, to be published, doi: 10.1109/TMECH.2021.3128992.
[32] J. Lee, K. Shin, and H. Jung, "Control scheme for rapidly responding register controller using response acceleration input in industrial roll-to-roll manufacturing systems," *IEEE Trans. Ind. Electron.*, vol. 69, no. 5, pp. 5215–5224, May 2022.
[33] Z. Chen, T. Zhang, Y. Zheng, D. S.-H. Wong, and Z. Deng, "Fully decoupled control of the machine directional register in Roll-to-Roll printing system," *IEEE Trans. Ind. Electron.*, vol. 68, no. 10, pp. 10007–10018, Oct. 2021.

**Jingyang Yan** received the B.S. degree in automation from Zhejiang University, Hangzhou, China, in 2014, and the M.S. degree in control engineering from the University of Science and Technology of China, Hefei, China, in 2017. He is currently working toward the Ph.D. degree in mechanical engineering in the University of Massachusetts, Amherst, MA, USA.

His research interests include deep learning, flexible electronics manufacturing, and nonlinear control.

**Xian Du** (Member, IEEE) received the B.S. degree in mechanical engineering from Tianjin University, Tianjin, China, in 1994, the M.S. degree in mechanical engineering from Shanghai Jiaotong University, Shanghai, China, in 2001, and the M.S. and Ph.D. degrees in the program of innovation of manufacturing systems and technology from the Singapore-MIT Alliance, National University of Singapore, Singapore, in 2002 and 2007, respectively.

Since 2018, he has been an Assistant Professor with the Mechanical and Industrial Engineering Department, Institute for Applied Life Sciences, University of Massachusetts, Amherst, MA, USA. His current research topic is the innovation of high-resolution, large-area, and fast-speed machine vision and pattern recognition technologies for manufacturing and medical devices. His research interests include pattern recognition, intelligent imaging and vision, flexible electronics manufacturing, robotics, and medical device realization.

Dr. Du was the recipient of the NSF CAREER award in 2020 and is also a member of the IEEE, ASME, and OSA.