

Synthesis of model predictive control and iterative learning control for topography regulation in additive manufacturing ^{*}

Zahra Afkhami ^{*} David Hoelzle ^{**} Kira Barton ^{*}

^{*} *Mechanical Engineering Department, University of Michigan, Ann Arbor, MI 48109, USA. bartonkl@umich.edu*

^{**} *Mechanical and Aerospace Engineering Department, The Ohio State University, Columbus, OH 43210, USA. hoelzle.1@osu.edu*

Abstract: This paper presents a spatially derived control solution for improving the performance of a high-resolution additive manufacturing (AM) process towards the fabrication of repeatable, thin-film functional devices. In particular, this work addresses challenges in fabrication of high-quality films using AM technology by incorporating knowledge about the interactions between printed layers of material within a control framework for improved repeatability and reliability in the fabrication of multi-layered micro devices using AM technology. We implement an SILC-MPC method that leverages the information from previous layers using spatial iterative learning control (SILC) and projects forward the data from future layers using model predictive control (MPC) to improve the tracking performance of iteration varying AM processes. Simulation results of an AM process termed electrohydrodynamic jet (e-jet) printing demonstrate that an SILC-MPC framework is effective and robust to repetitive and nonrepetitive model uncertainties and outperforms traditional SILC by converging faster to the nominal behavior with a lower tracking error.

Copyright © 2022 The Authors. This is an open access article under the CC BY-NC-ND license (<https://creativecommons.org/licenses/by-nc-nd/4.0/>)

Keywords: Model predictive control, Iterative learning control, Micro-Additive Manufacturing .

1. INTRODUCTION

Additive manufacturing (AM), also known as 3D printing, is a class of processes that enable fabrication of 3D structures through selective addition of material in a layer-by-layer manner. AM processes are becoming more popular, because they are fast, cost effective, and do not need harsh chemical processes (Afkhami et al. (2020a); Balta et al. (2019)). Most AM processes, and in particular μ -AM processes, run in open-loop with system parameters that are tuned by human operators through trial and error. In μ -AM systems, such as electrohydrodynamic jet (e-jet) printing, temporal events occur on the order of *ms* with spatial resolutions as small as $1\mu\text{m}$ or less. Despite the many advantages of AM, the lack of real-time monitoring methods that can capture, analyze, and react to *in situ* measurements has been a challenge for most AM processes and in particular μ -AM systems, such as e-jet printing (Mohammadi et al. (2019); Rose et al. (2021); Spiegel et al. (2019); Landers et al. (2020)), in which the key dynamics occur at the micro-/nano- scales. Due to time- and length-scale limitations of μ -AM processes, online measurement of system parameters is challenging; therefore, traditional temporal feedback control is not a good candidate to direct the material deposition of these processes.

In many AM structures, such as the optical sensor presented in Afkhami et al. (2020a), spatial factors such as droplet coalescence or film flatness and consistency play a larger role in device performance than temporal events. Therefore, for these systems, the spatial tracking errors $e(x, y)$ are generally more important than temporal errors $e(t)$. To address this performance objective, spatial control frameworks like spatial iterative learning control (SILC) provide a suitable framework for regulating the material deposition of these systems (Altin et al. (2018); Hoelzle and Barton (2016, 2014); Afkhami et al. (2020b); Aarnoudse et al. (2019); Wang et al. (2016b)). SILC is an extension of temporal ILC (Bristow et al. (2006); Norrlöf and Gunnarsson (2002); Meng and Moore (2017)) reformatted for the spatial domain. SILC uses spatial data (e.g. layer height) from previous iterations to derive a feedforward control signal for the next printed iteration. In addition to the layer to layer dynamics, it is important to consider process constraints within the AM system, such as the requirement of strictly positive control inputs, $f(x, y)$. Model predictive control (MPC) is a control framework that works especially well for constrained systems and incorporates a forward projection process that is useful for AM systems in which the current layers directly impact future layers. MPC leverages the system model and the data from past iterations to predict optimized control actions for multiple steps ahead in the future (Rawlings et al. (2017); Li et al. (2019); Lee et al. (1999); Xie and Ren (2018); Rosolia and Borrelli (2017); Gegel et al. (2019)). MPC has been already applied to AM process to control 1D height increment with varying reference (Gegel et al.

^{*} This material is based upon work supported by the National Science Foundation under Grant No. CMMI-1727894. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the authors and do not necessarily reflect the views of the National Science Foundation.

(2019)). Similarly, predictive iterative learning control in and Yu and Hou (2016); Wang et al. (2016a) has a similar idea that uses past information and a prediction of future iterations to update its control input.

Through AM, a 3D structure is fabricated by repetitive addition of material. The behavior of a printed layer depends on interactions between the printed material and the topography from previous layers. This results in an iteration varying system in which the system parameters and plant dynamics, $\mathbf{H}$, change from layer to layer. The ability to achieve robustness through control design highly depends on the magnitude of the plant model variations, $\Delta\mathbf{H}$. To tackle the iteration varying behavior of many AM processes and enable robust control designs, the system dynamics can be formulated to ensure robustness under interval uncertainty assuming plant model variations are bounded (Altin et al. (2018); Ahn et al. (2007)). In our previous work (Afkhami et al. (2021)), we applied higher-order SILC (HO-SILC) (similar to Fig. 1a) to a model of an AM process. This higher-order controller leveraged data from previous devices and previous layers within the same device to construct an optimized input for the next layer of a 3D structure. We term this type of learning controller as backward learning. A drawback of this previously presented HO-SILC framework was that it was overly conservative for systems with minimal model uncertainty since the controller was designed to ensure robustness for the maximum amount of uncertainty, thus causing the system to lose performance advancements.

To achieve a less restrictive bound on the prediction horizon, researchers have introduced a finite-tail MPC formulation that solves a constrained optimal control problem based on prediction and optimization. MPC exploits the system model to predict performance error signals for multiple steps (layers) in the future and consequently make informed control decisions that work to achieve performance gains across all of the layers. This approach is known as a receding horizon strategy in which a sequence of input signals for multiple steps (layers) in the future is planned, and yet only the next control action is applied. The sequence of control inputs will be recomputed once new information is available. Although closed-loop stability is not generally guaranteed with an MPC framework, adding a final/terminal cost can improve the stability properties through improvements in convergence speed and tracking error. Stochastic and adaptive MPC provide alternative techniques that have been used to achieve these desirable properties (Sun et al. (2013); Bichi et al. (2010)).

The work presented in this paper investigates whether the integration of SILC with MPC into a combined learning control framework will result in a layer-to-layer process that is more robust to model uncertainty compared to traditional SILC, and yet maintains the high performance advancements expected from these control architectures. Note that in the multi-layer structures considered in this work, the layers are built on top of each other, such that the printing behavior for a current layer depends directly on the topography of the previous layers. The goal of this work is to derive an integrated control framework that leverages the information from previous layers using HO-SILC, plus a prediction from future layers using MPC (see Fig. 1b), to minimize the total error and individual layer

errors within an AM process simultaneously, while achieving uniform thin-film fabrication with minimal surface roughness. The novel contributions of this work include:

- Derivation of a higher-order SILC framework for minimizing layer and total errors simultaneously (Fig. 1a) to more effectively fabricate 3D structures in a layer-by-layer fashion.
- Development of a combined spatial ILC and terminal cost MPC framework that incorporates backward learning through SILC with forward learning using MPC to improve the performance of AM processes.
- Demonstration and analysis of the performance advancements achieved through the implementation of the combined learning control framework on a simulated model of a high-resolution printing process for the fabrication of 3D constructs.

The remainder of this paper is organized as follows. The e-jet system and the layer-to-layer material addition process is detailed in Section 2. The mathematical derivations of the SILC and SILC-MPC controllers for AM structures are detailed in Section 3. In Section 4, the numerical simulation results for a multi-layer e-jet printed structure are presented and the performance of SILC and SILC-MPC controllers are compared. Finally, the conclusions are given in Section 5. Complementary material is presented in Appendix.

2. AM PROBLEM FORMULATION

The proposed learning control framework is designed for additive manufacturing processes. In particular, this work considers a model of a μ -AM process, known as Electrohydrodynamic jet (e-jet) printing. E-jet printing is a μ -AM technique that enables fabrication of 3D components with micro-scale spatial and nano-scale thickness resolution. Figure 2B shows a multi-material e-jet printer located at the University of Michigan. The main elements of this e-jet printer are a conductive nozzle, grounded substrate, translational stage, ink chamber containing liquid ink, and an integrated atomic force microscopy (AFM) for measurement of layer topography. A high voltage pulse with short pulsewidth, t_p , is applied at the nozzle tip as shown in Fig. 2A, pulling the liquid ink from the nozzle to the substrate. By synchronizing the stage motion with the applied voltage at each spatial location, drop-on-demand printing is achieved with the ability to deposit a droplet at a desired location. By placing droplets adequately close to each other with fixed droplet-to droplet distance known as pitch, uniform thin-films such as the one in Fig. 2C-E are fabricated (Afkhami et al. (2020a)). The sequence of material addition and heightmap evolution of multi-layer structures in an additive process is described as the following 2D convolution equation (Pannier et al. (2019)):

$$\Delta g_l(x, y) \triangleq \sum_{m \in \mathbb{Z}_M, n \in \mathbb{Z}_N} h_{l-1}^{(m, n)}(x - m, y - n) * f_l(m, n), \quad (1)$$

where $g_l(x, y) \in \mathbb{R}^{M \times N}$ and $f_l(x, y) \in \mathbb{R}^{M \times N}$ are the deposited topography (output) and cube root of droplet volume (input) at each spatial location (x, y) at layer l , respectively. Droplet volume is controlled through the applied pulsewidth, t_p . Topography increment at layer l is defined as, $\Delta g_l(x, y) \triangleq g_l(x, y) - g_{l-1}(x, y)$. The response

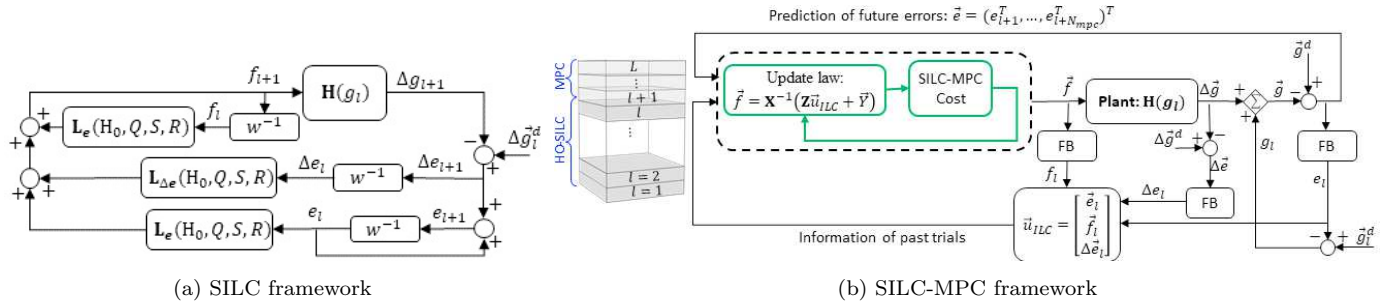


Fig. 1. **Block diagrams of SILC and SILC-MPC frameworks:** Learning occurs in the vertical direction. A) SILC utilizes the knowledge from previous layers to predict the optimized input of the next layer. The knowledge here refers to the input signal, f_l , layer (or incremental) error, Δe_l , and total error, e_l at layer l , which is sum of the incremental errors over all layers, $e_l = \sum_{i=1}^l \Delta e_i$. B) Learning occurs from previous layers using SILC design, as well as a prediction of future layer depositions using MPC design, by incorporating a prediction of future layer errors, e_{l+n} , to predict the input signal of multiple layers ahead, $\vec{f} = (f_{l+1}, \dots, f_{l+N_{mpc}})$. At each printing pass, only the input signal of the first layer, f_{l+1} , is considered. Here, each layer is one iteration and w^{-1} is the trial-delay operator. $\vec{e} = (\vec{e}_{l+1}^T, \dots, \vec{e}_{l+N_{mpc}}^T)^T$, $\vec{g} = (\vec{g}_{l+1}^T, \dots, \vec{g}_{l+N_{mpc}}^T)^T$, $\Delta \vec{e} = (\Delta \vec{e}_{l+1}^T, \dots, \Delta \vec{e}_{l+N_{mpc}}^T)^T$, and $\Delta \vec{g} = (\Delta \vec{g}_{l+1}^T, \dots, \Delta \vec{g}_{l+N_{mpc}}^T)^T$. FB operator takes a stacked vector, (for example $\vec{e}$), extracts its first block, $\vec{e}_{l+1}$, and applies the shift-delay operator to extract $\vec{e}_l$.

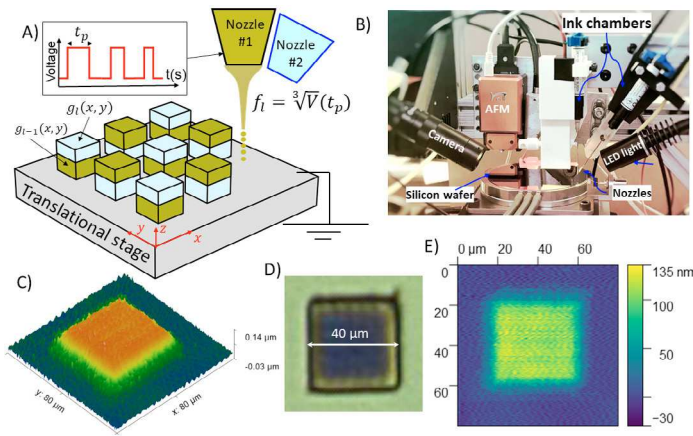


Fig. 2. **High-resolution fabrication using e-jet printing technology.** A) schematic of an e-jet printer with dual nozzles. B) e-jet Printer testbed located at the University of Michigan. C), D) and E) AFM scans and optical microscopy of a $40 \times 40 \mu\text{m}^2$ thin-film pattern deposited by the e-jet system in Fig 2B.

of the build material to an impulse applied at coordinate (m, n) and layer $l-1$ is given as $h_{l-1}^{m,n}(x, y)$, which describes the spatial distribution and spreading of the droplet into nearby pixels. To better represent the printing environment, a normally distributed iteration-varying white noise signal, Δh_l , is added to the plant model, with a mean and variance of 0.00 and 0.2, respectively. Repetitive model uncertainty is also added to represent unmodeled surface variations or initial tilts within the system and is represented by $\Delta h_{rep} = 0.1 * h_0$. Combining these features together, the iteration varying spatial impulse response used for (1) will be $h_l = h(g_l) + \Delta h_l + \Delta h_{rep}$. Note that the plant matrix, $h(g_l)$, is a function of previous layer topography. We use the methods described in Pannier et al. (2019); Afkhami et al. (2021, 2020b) to calculate the impulse response for a given layer. In order to develop the control framework, the 2D convolution in (1) needs to be converted to the following input-output lifted- version,

$$\Delta \vec{g}_l = \mathbf{H}(g_{l-1}) \vec{f}_l \quad (2a)$$

$$\vec{g}_l = \sum_{i=1}^l \Delta \vec{g}_i = \sum_{i=1}^l \mathbf{H}(g_{i-1}) \vec{f}_i \quad (2b)$$

where $\vec{g}_l \triangleq \nu(g_l(x, y)) \in \mathbb{R}^{MN \times 1}$ and $\vec{f}_l \triangleq \nu(f_l(x, y)) \in \mathbb{R}^{MN \times 1}$ represent the vectorized forms of the output and input signals through the use of $\nu(\cdot)$ as a vectorization operator that converts a matrix, $P \in \mathbb{R}^{m \times n}$, into a column vector form, $\vec{P} \in \mathbb{R}^{mn \times 1}$. $\mathbf{H}(g_{l-1}) \in \mathbb{R}^{MN \times MN}$ is the iterative plant matrix associated with the iteration varying impulse response $h_{l-1}^{(m,n)}$, and $\mathbf{H}(g_0)$ is the plant matrix associated with the first layer deposited on the substrate, which we assume to be a smooth pre-layer of cured ink. In addition, the incremental layer-error ($\Delta \vec{e}_l$) and total build error ($\vec{e}_l$) are defined as,

$$\begin{aligned} \Delta \vec{e}_l &\triangleq \Delta \vec{g}_l^d - \Delta \vec{g}_l = \Delta \vec{g}_l^d - \mathbf{H}_{l-1} \vec{f}_l, \\ \vec{e}_l &\triangleq \vec{g}_l^d - \vec{g}_l = \sum_{i=1}^l \Delta \vec{e}_i, \quad \vec{g}_l^d = \sum_{i=1}^l \Delta \vec{g}_i^d \end{aligned} \quad (3)$$

where $\Delta \vec{g}_l^d$ and $\vec{g}_l^d$ are the desired incremental and total build height maps at layer l .

3. CONTROLLER FORMULATION

In our previous work (Afkhami et al. (2021)), we used a backward learning method through an SILC design similar to Fig. 1a that leveraged the data from previous iterations to construct the optimized input. From a manufacturing perspective, our goal was to develop a control framework that regulates the deposition process to realize a vertically stacked multi-layer/multi-material structure with consistent layer thickness, which is a highly desirable characteristic in many sensory applications, such as optical sensors where uniformity and periodicity of the layers are of great importance Afkhami et al. (2020a); Lequime et al. (2015). In these examples, the norm optimal SILC (NO-SILC) cost function was defined based on the layer errors, $\Delta \vec{e}_l$. The proposed SILC resulted in consistent layer deposition with $\Delta \vec{e}_l$ converging from layer-to-layer, resulting in uniform layers. However, SILC did not weight the total build error $\vec{e}_l$, resulting in the total error increasing over the layers. In AM structures, the layer errors add up during the build and may result in a large deviation from the desired height, especially when the layer number is large. Therefore, if the printing objective requires total device dimension convergence as well as layer-by-layer convergence, minimizing

the layer errors alone may not be sufficient. To address this need, the work in this paper investigates a modified backward learning controller that incorporates individual layer as well as total device errors combined with a forward learning approach that uses MPC to achieve a projection in the printing process to enhance the control decisions. The combined control architecture can be found in Fig .1.

3.1 SILC Framework

In this section, we use a norm-optimal SILC (NO-SILC) framework to design robust learning filters. Assuming a constant desired height for all layers, we modify our previous cost function in Afkhami et al. (2022) to minimize the total error and layer error simultaneously.

$$\mathcal{J}^{ILC} = \Delta \vec{e}_{l+1}^T \mathbf{Q}_{\Delta e} \Delta \vec{e}_{l+1} + \vec{e}_{l+1}^T \mathbf{Q}_1 \vec{e}_{l+1} + \vec{f}_{l+1}^T \mathbf{S}_1 \vec{f}_{l+1} + (\vec{f}_{l+1} - \vec{f}_l)^T \mathbf{R}_1 (\vec{f}_{l+1} - \vec{f}_l), \quad (4)$$

where $\mathbf{Q}_{\Delta e}$, $\mathbf{Q}_1$, $\mathbf{S}_1$, $\mathbf{R}_1$ are weighting matrices used to weight the layer errors, total errors, control inputs and change in control inputs. These matrices are generally defined as identity matrices scaled by non-negative scalars such that $\mathbf{Q}_{\Delta e} = q_{\Delta e} \mathbf{I}$, $\mathbf{Q}_1 = q_1 \mathbf{I}$, $\mathbf{S}_1 = s_1 \mathbf{I}$, $\mathbf{R}_1 = r_1 \mathbf{I}$. The optimal feedforward signal is achieved by solving the following optimization problem,

$$\frac{\partial \mathcal{J}^{ILC}}{\partial \vec{f}_{l+1}} = -\mathbf{H}_l^T \mathbf{Q}_1 (\vec{e}_l + \Delta \vec{e}_l + \mathbf{H}_{l-1} \vec{f}_l - \mathbf{H}_l \vec{f}_{l+1}) + \mathbf{S}_1 \vec{f}_{l+1} + \mathbf{R}_1 (\vec{f}_{l+1} - \vec{f}_l) - \mathbf{H}_l^T \mathbf{Q}_{\Delta e} (\Delta \vec{e}_l + \mathbf{H}_{l-1} \vec{f}_l - \mathbf{H}_l \vec{f}_{l+1}) = 0. \quad (5)$$

Solving for the updated control signal, the SILC update law and learning filters can be shown to simplify to:

$$\vec{f}_{l+1} = L_f \vec{f}_l + L_e \vec{e}_l + L_{\Delta e} \Delta \vec{e}_l \quad (6a)$$

$$f_{l+1}(u, v) = (L_f \circ f_l + L_e \circ e_l + L_{\Delta e} \circ \Delta e_l)(u, v) \quad (6b)$$

where $\circ$ is the Hadamard or entrywise product of two matrices. Assuming $\mathbf{H}$ can be written as block circulant with circulant blocks (BCCB) (Hoelzle and Barton (2016)), (6b) is the equivalent frequency-domain representation of (6a). BCCB matrices are defined in more detail in Hoelzle and Barton (2016).

The control update, $f_{l+1}(u, v)$, is the 2D discrete Fourier transform (DFT) representation of $f_{l+1}(x, y)$. More details can be found in Hoelzle and Barton (2016). Note that the update law in (6) is higher order because the total error in a multi-layer structure is a function of the incremental errors from all previous layers, $e_l = \sum_{i=1}^l \Delta e_i$. Importantly, if one wanted to design an iteration invariant controller, the learning filters must be based on the nominal plant $\mathbf{H}_0$ from (5) as demonstrated below:

$$L_f(u, v) = \frac{(q_1 + q_{\Delta e})H_0^* \circ H_0 + r_1}{(q_1 + q_{\Delta e})H_0^* \circ H_0 + (s_1 + r_1)}(u, v) \quad (7a)$$

$$L_e(u, v) = \frac{q_1 H_0^*}{(q_1 + q_{\Delta e})H_0^* \circ H_0 + (s_1 + r_1)}(u, v) \quad (7b)$$

$$L_{\Delta e}(u, v) = \frac{(q_1 + q_{\Delta e})H_0^*}{(q_1 + q_{\Delta e})H_0^* \circ H_0 + (s_1 + r_1)}(u, v) \quad (7c)$$

3.2 SILC-MPC Framework

In this section, we extend the higher-order SILC framework in Section 3.1 to an SILC-MPC framework that leverages the information from previous layers, as well as

a prediction of future layers to determine an optimized feedforward input signal.

The quadratic cost function for the combined controller is defined as,

$$\mathcal{J} = \mathcal{J}^{ILC} + \mathcal{J}^{MPC} = \vec{e}^T \mathbf{Q} \vec{e} + \vec{f}^T \mathbf{S} \vec{f} + \Delta \vec{f}^T \mathbf{R} \Delta \vec{f} \quad (8a)$$

$$\mathcal{J}^{MPC} = \mathcal{J}_t + \sum_{j=2}^{N_{mpc}} \mathcal{J}_{l+j} \\ \mathcal{J}_{l+j} = \vec{e}_{l+j}^T \mathbf{Q}_j \vec{e}_{l+j} + (\vec{f}_{l+j} - \vec{f}_{l+j-1})^T \mathbf{R}_j (\vec{f}_{l+j} - \vec{f}_{l+j-1}) + \vec{f}_{l+j}^T \mathbf{S}_j \vec{f}_{l+j}, \quad (8b)$$

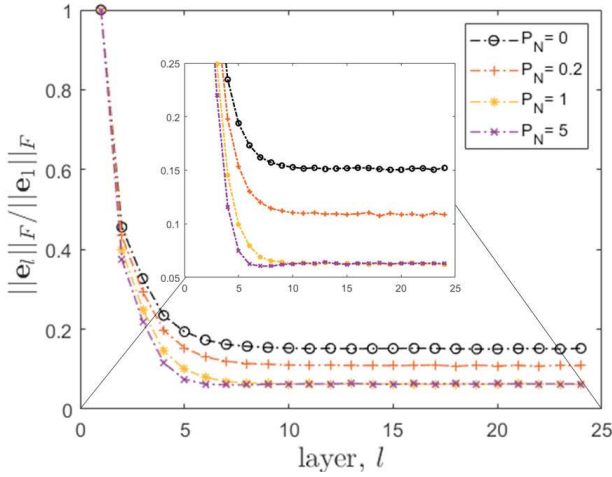
where $\mathbf{Q}_j$, $\mathbf{S}_j$, $\mathbf{R}_j$ are weighting matrices that are defined as $\mathbf{Q}_j = q_j \mathbf{I}$, $\mathbf{S}_j = s_j \mathbf{I}$, and $\mathbf{R}_j = r_j \mathbf{I}$, and q_j , s_j , and r_j are positive scalars. N_{mpc} denotes the MPC horizon. The other components in (8) are defined as:

$$\mathbf{Q} = \begin{bmatrix} \mathbf{Q}_{\Delta e} & \dots & 0 \\ 0 & \mathbf{Q}_1 & \vdots \\ \vdots & \ddots & 0 \\ 0 & \dots & \mathbf{Q}_{N_{mpc}} + \mathbf{P}_N \mathbf{I} \end{bmatrix}, \quad \vec{e} = \begin{bmatrix} \Delta \vec{e}_{l+1} \\ \vec{e}_{l+1} \\ \vec{e}_{l+2} \\ \vdots \\ \vec{e}_{l+N_{mpc}} \end{bmatrix}, \\ \vec{f} = \begin{bmatrix} \vec{f}_{l+1} \\ \vec{f}_{l+2} \\ \vdots \\ \vec{f}_{l+N_{mpc}} \end{bmatrix}, \quad \mathbf{S} = \begin{bmatrix} \mathbf{S}_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \mathbf{S}_{N_{mpc}} \end{bmatrix} \\ \Delta \vec{f} = \begin{bmatrix} \vec{f}_{l+1} - \vec{f}_l \\ \vec{f}_{l+2} - \vec{f}_{l+1} \\ \vdots \\ \vec{f}_{l+N_{mpc}} - \vec{f}_{l+N_{mpc}-1} \end{bmatrix}, \quad \mathbf{R} = \begin{bmatrix} \mathbf{R}_1 & \dots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \dots & \mathbf{R}_{N_{mpc}} \end{bmatrix}.$$

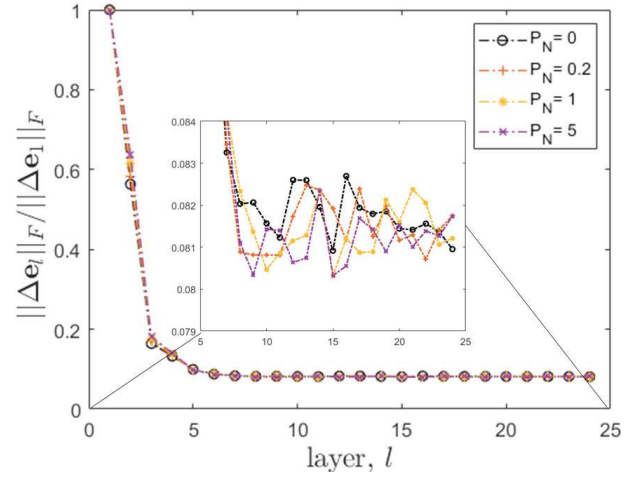
The MPC cost function in (8b) contains two elements. The first element is the terminal cost, denoted by $\mathcal{J}_t$, that ensures closed-loop stability of the MPC algorithm (Rawlings et al. (2017); Li et al. (2019)). The second term is a projection forward in the layer direction that aims to minimize the total error in future layers by predicting the system behavior in future layers and using it to determine the control input for the next layer. We define the terminal cost as $\mathcal{J}_t \triangleq \vec{e}_{l+N_{mpc}}^T (\mathbf{P}_N \mathbf{I}) \vec{e}_{l+N_{mpc}}$, where $\mathbf{P}_N$ is a positive scalar. The MPC cost function starts from $j = 2$, since $j = 1$ is already taken into account in the HO-SILC algorithm provided in (4). Let $\vec{u}_{ILC}$ denote a vector that contains the information from previous layers, such that $\vec{u}_{ILC} = [\vec{e}_l^T, \vec{f}_l^T, \Delta \vec{e}_l^T]^T$. To find the optimal control sequence $\vec{f}^*$, the cost function in (8) is differentiated with respect to $\vec{f}$ and equated to zero.

$$\nabla_{\vec{f}} \mathcal{J} = \left[\frac{\partial \mathcal{J}^T}{\partial \vec{f}_{l+1}}, \dots, \frac{\partial \mathcal{J}^T}{\partial \vec{f}_{l+N_{mpc}}} \right]^T = \mathbf{X} \vec{f} + \mathbf{Z} \vec{u}_{ILC} + \vec{Y}, \quad (9)$$

with $\mathbf{X} \in \mathbb{R}^{MNN_{mpc} \times MNN_{mpc}}$, $\mathbf{Z} \in \mathbb{R}^{MNN_{mpc} \times 3MN}$, and $\vec{Y} \in \mathbb{R}^{MNN_{mpc} \times 1}$. An analytical expression for $\mathbf{X}$, $\mathbf{Z}$ and $\vec{Y}$ can be achieved by deriving the partial derivatives given in (9) using the cost function given in (8). From these derivations, we can determine that the structure of $\mathbf{X}$, $\mathbf{Z}$ and $\vec{Y}$ has the following relationship:



(a) **Normalized total error:** higher vales of P_N lead to decreases in total error. We fix $P_N = 1$, since convergence properties do not change for higher values of $P_N > 1$.



(b) **Normalized layer error:** varying values of P_N appear to have no effect on the incremental error. Note that terminal cost is only imposed on the total error in (8b).

Fig. 3. Effect of terminal cost on convergence properties of SILC-MPC: The results are shown for SILC-MPC with $N_{mpc} = 2$. Adding a terminal cost in (8b) improves the SILC-MPC performance for total error with minimal impact on layer error.

$$\begin{aligned}
 X_{i,i} &= \mathbf{H}_0^T (P_N \mathbf{I} + \mathbf{Q}_{\Delta e} \Delta(i, 1) + \sum_{n=i}^{N_{mpc}} \mathbf{Q}_n) \mathbf{H}_0 + (\mathbf{R}_i + \mathbf{S}_i + \mathbf{R}_{i+1}), \\
 X_{i,j} &= -\mathbf{H}_0^T (P_N \mathbf{I} + \sum_{n=k}^{N_{mpc}} \mathbf{Q}_n) \mathbf{H}_0 - \mathbf{R}_k, \quad j = i+1 \text{ or } j = i-1 \\
 X_{i,j} &= -\mathbf{H}_0^T (P_N \mathbf{I} + \sum_{n=k}^{N_{mpc}} \mathbf{Q}_n) \mathbf{H}_0, \quad j < i-1 \text{ or } j > i+1
 \end{aligned} \tag{10}$$

where $k = \max(i, j)$ and $\Delta(i, 1) = 1$ if $i = 1$ and is zeros otherwise. Similarly $\mathbf{Z}$ has the following format

$$\begin{aligned}
 Z_{i,1} &= -\mathbf{H}_0^T (P_N \mathbf{I} + \sum_{n=i}^{N_{mpc}} \mathbf{Q}_n) \\
 Z_{1,2} &= -\mathbf{H}_0^T (\mathbf{Q}_1 + \mathbf{Q}_{\Delta e}) \mathbf{H}_0 - \mathbf{R}_1 \\
 Z_{1,3} &= -\mathbf{H}_0^T (\mathbf{Q}_1 + \mathbf{Q}_{\Delta e}) \\
 Z_{i,2} &= Z_{i,3} = \mathbf{0} \quad i > 1
 \end{aligned} \tag{11}$$

Likewise $\vec{Y}$ has the following format:

$$\begin{aligned}
 \vec{Y}(1) &= -\mathbf{H}_0^T (N_{mpc} P_N \mathbf{I} + \sum_{n=2}^{N_{mpc}} (n \mathbf{Q}_n)) \vec{r} \\
 \vec{Y}(i) &= -\mathbf{H}_0^T (N_{mpc} P_N \mathbf{I} + \sum_{n=i}^{N_{mpc}} (n \mathbf{Q}_n)) \vec{r}, \quad i > 1
 \end{aligned} \tag{12}$$

Assuming $\mathbf{X}$ is invertible, the optimal control input $\vec{f}^*$ is then obtained by solving $\mathbf{X} \vec{f}^* + \mathbf{Z} \vec{u}_{ILC} + \vec{Y} = \vec{0}$ as $\vec{f}^* = -\mathbf{X}^{-1}(\mathbf{Z} \vec{u}_{ILC} + \vec{Y})$. We only implement the first block of $\vec{f}^*$, which is related to the control input $\vec{f}_{l+1}^*$ of the next layer. The derivation of (10-12) is shown in the appendix. Note that although $\mathbf{X}$ is symmetric, and each block matrix of $\mathbf{X}$ and $\mathbf{Z}$ is BCCB, the overall $\mathbf{X}$ and $\mathbf{Z}$ matrices are not BCCB; thus, we have to invert a high dimensional matrix, $\mathbf{X}$, to calculate the optimal solution $\vec{f}^*$. It is important to note that this calculation for the SILC-MPC controller will be computationally more expensive than the update law required for SILC in (6) and (7). Care must be taken to determine when this approach is necessary and the computation burden is acceptable for the given system requirements. Future work

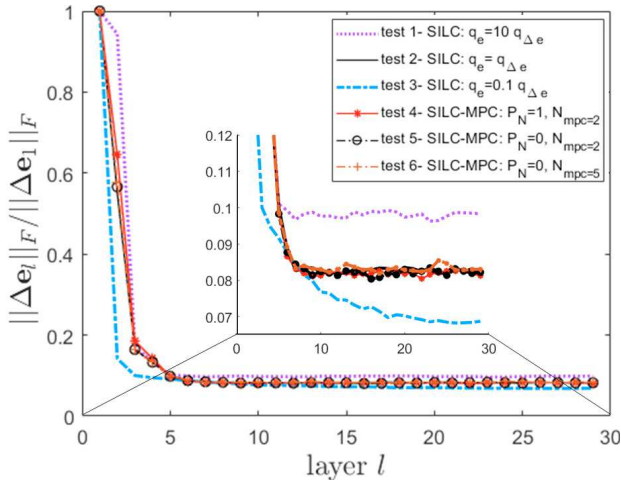
will investigate methods to reformulate the SILC-MPC update law in a way that the DFT computations are possible and will result in a reduced computational burden. In addition, future work will focus on the relationship between the invertibility condition of the $\mathbf{X}$ matrix and stability and convergence of the SILC-MPC algorithm.

4. SIMULATION VALIDATION

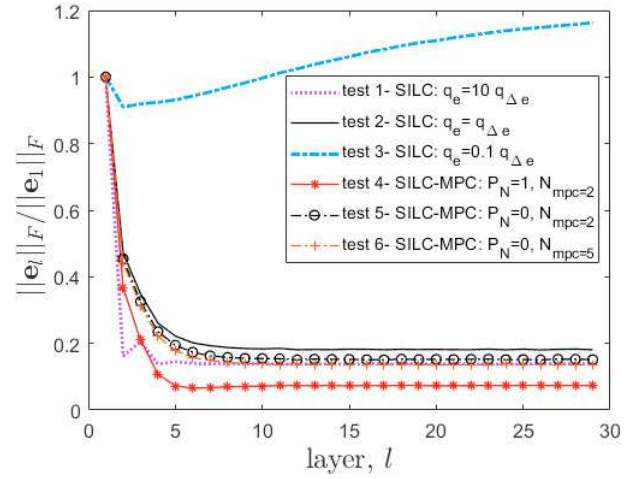
In this section, simulation results for the system described in Section 2 controlled by the SILC-MPC and SILC frameworks are presented. As described in Fig. 1b, the SILC-MPC controller uses the information from previous layer depositions ($\mathbf{f}_l$, $\mathbf{e}_l$, and $\Delta \mathbf{e}_l$) through an SILC design, and leverages a prediction of what will happen in future depositions ($\mathbf{e}_{l+2}, \dots, \mathbf{e}_{N_{mpc}}$) through an MPC design, to estimate an optimized feedforward signal for future layer depositions, $\mathbf{f}_{l+1}, \dots, \mathbf{f}_{l+N_{mpc}}$. However, at each printing pass, we only implement the current layer input signal $\mathbf{f}_{l+1}$, recalculating the feedforward signals each layer. A multi-layer structure similar to Fig. 2 with $100\mu\text{m} \times 100\mu\text{m}$ spatial resolution, 150 nm layer thickness resolution, and 25 layers is considered. For the first iteration, $l = 1$, we use a nonzero input signal such that the first layer thickness was around 130 ± 22 nm, which is computed based on knowledge of conventional pre-prints.

To design the weighting parameters for the SILC-MPC cost function in (8), we define the following vectors, $\vec{q} = [q_{\Delta e}, q_1, \dots, q_{N_{mpc}}]^T$, $\vec{s} = [s_1, \dots, s_{N_{mpc}}]^T$, $\vec{r} = [r_1, \dots, r_{N_{mpc}}]^T$. The weighting coefficients for the two frameworks are presented in test1-test3 and test4-test6 of Table 1, respectively. The weighting parameters are designed to be equally weighted such that $\|\vec{q}\| = 1$, $\|\vec{s}\| = 0.03$, $\|\vec{r}\| = 0.01$, which are the same weighting parameters that we used in our prior work Afkhami et al. (2022).

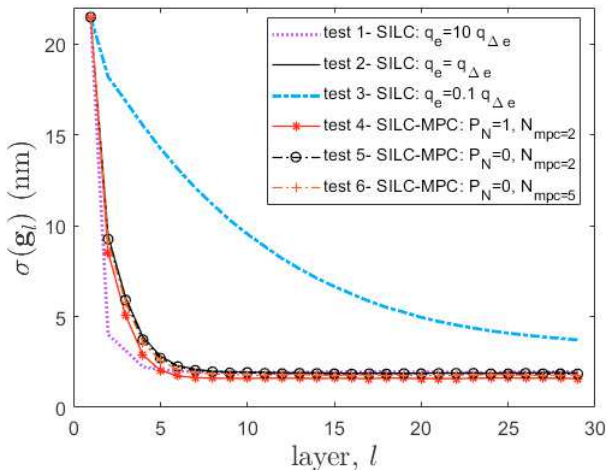
Figure 3 shows the effect of terminal cost in (8) on the SILC-MPC performance. As demonstrated in MPC literature (Lee et al. (1999); Pannek and Grüne (2011); Xie and Ren (2018)), adding a terminal cost improves the convergence and stability properties of the SILC-MPC



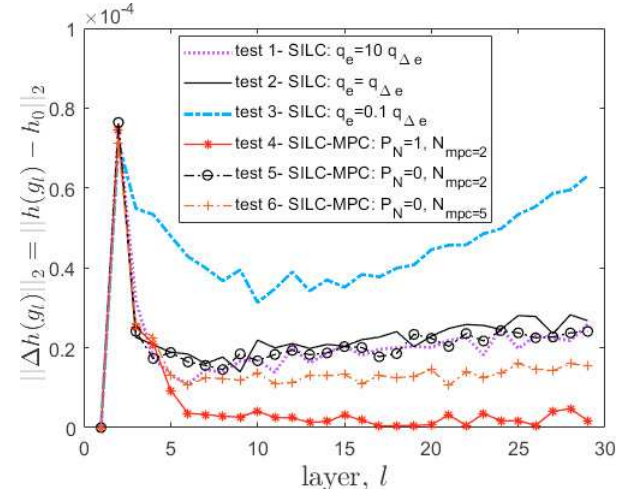
(a) **Normalized layer error:** Test 3 with the largest $q_{\Delta e}$ gain (smallest q_e) (see Table 1) converges faster to a lower final value for layer error. If repeatable layer deposition is desirable, the SILC controller in test 3 is a good choice.



(b) **Normalized total error:** similar to the results in Afkhami et al. (2022), total error increases over the build if layer error is emphasized. SILC-MPC with a terminal cost (test4) results in the lowest combined layer and total errors.



(c) **Surface roughness:** roughness, which is an indication of layer flatness, decreases over the layers for all tests. Importantly, lower roughness values result in smaller deviations from the nominal plant $h_l(x, y) \rightarrow h_0$, which is confirmed in Fig 4d.



(d) **Simulation model uncertainty:** test 4 that includes a terminal cost results in the smallest differences between the nominal and actual plant matrices.

Fig. 4. SILC (tests 1-3) and SILC-MPC (tests 4-6) convergence comparison. Note that for both controllers the weighting coefficients are chosen such that $\|\vec{q}\| = 1$, $\|\vec{r}\| = 0.01$, $\|\vec{s}\| = 0.03$. a) normalized layer error versus layer number, b) normalized total error versus layer index, c) standard deviation of total height versus layer index, d) model uncertainty calculated by the difference between $h(g_l)$ and h_0 derived using the M3 method from Pannier et al. (2019) for NOA170 in Afkhami et al. (2022).

algorithm with respect to total error and can regulate both layer and total errors at the same time. Next, we set $P_N = 1$ and investigate the effect of mpc horizon on the controller performance.

The performance of the SILC and SILC-MPC controllers is compared in Fig. 4 and Table 1. Tests 1-3 are related to the SILC framework with $\vec{q} = [q_{\Delta e}, q_1 = q_e]^T$, $\vec{s} = [0.03]$, $\vec{r} = [0.01]$, while tests 4-6 are related to the SILC-MPC framework. The simulation results imply that there is a trade-off between performance and computation time of the SILC-MPC controller. The SILC-MPC framework generally has lower total error and surface roughness compared to the SILC controller and can regulate both layer and total error at the same time. However, the SILC-MPC framework is computationally more expensive than the SILC controller. Based on the information in the

last column of Table 1, the simulation time for updating the feedforward signal per layer for the SILC controller is 50% lower as compared to the SILC-MPC controller with a terminal cost. Comparing tests 5 and 6, it is observed that although higher values of the receding horizon, N_{mpc} , enhance the SILC-MPC performance, the update-law takes more time to update the input signal. The reasons for these differences are that 1) matrices $\mathbf{X}$, $\mathbf{Z}$ and $\vec{Y}$ are high dimensional compared to the SILC learning filters $\mathbf{L}_f$, and $\mathbf{L}_e$, 2) the structure of $\mathbf{H}_0$ allows for the use of fast DFT computations in (7) that do not require inverse matrix calculations, while in the SILC-MPC update law, $\vec{f}^* = -\mathbf{X}^{-1}(\mathbf{Z} \vec{u}_{ILC} + \vec{Y})$, $\mathbf{X}$ is not BCCB and cannot use DFT calculations.

Similar to the traditional MPC in Rawlings et al. (2017); Pannek and Grüne (2011); Vallon and Borrelli (2020);

Table 1: SILC and SILC-MPC parameters and simulation time

		$\vec{q}$	$\vec{s}$	$\vec{r}$	P_N	N_{mpc}	time(s) /layer
SILC	test 1	$\frac{1}{\sqrt{26}} \begin{bmatrix} 1 \\ 5 \end{bmatrix}$	0.03	0.01	—	—	1.22
	test 2	$\frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix}$	0.03	0.01	—	—	1.22
	test 3	$\frac{1}{\sqrt{26}} \begin{bmatrix} 5 \\ 1 \end{bmatrix}$	0.03	0.01	—	—	1.22
SILC-MPC	test 4	$\frac{1}{\sqrt{3}} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$	$\frac{0.03}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix}$	$\frac{0.01}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix}$	1	2	2.59
	test 5	$\frac{1}{\sqrt{3}} \begin{bmatrix} 1 \\ 1 \\ 1 \end{bmatrix}$	$\frac{0.03}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix}$	$\frac{0.01}{\sqrt{2}} \begin{bmatrix} 1 \\ 1 \end{bmatrix}$	0	2	2.59
	test 6	$\frac{1}{\sqrt{6}} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$	$\frac{0.03}{\sqrt{5}} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$	$\frac{0.01}{\sqrt{5}} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix}$	0	5	20.9

Rosolia and Borrelli (2017), simulation results in Fig. 4b show that as the MPC horizon increases, the performance of an SILC-MPC control law without the terminal cost (tests 5 and 6, $P_N = 0$) approaches an SILC-MPC with terminal cost (test 4, $P_N = 1$). Comparing the simulation results in Figures 4a-4d, the SILC-MPC controller with a terminal cost formulation in test 4 provides a more robust and stable controller that results in decreases to the layer and total errors simultaneously, leading to highly uniform stacked thin-films with low surface roughness.

5. CONCLUSION

In this work, we present a novel spatial topography controller to enable enhanced deposition at the micro-scale. The framework combines spatial iterative learning control with model predictive control to achieve enhanced performance and robustness. The proposed controller is robust to stochastic and repetitive model uncertainty. Simulation results of an e-jet printed structure controlled by the SILC-MPC framework indicate that high-quality thin-films with uniform and repeatable thickness resolutions are achievable by tuning the MPC terminal cost and ILC parameters. The results imply that SILC-MPC framework generally has better performance compared to a traditional SILC controller. However, the SILC update law is much faster (50%) compared to the SILC-MPC algorithm. Future work will consider non-negativity constraints in the input along with experimental validations. In addition, reformulation of the SILC-MPC update law in a way that DFT computations are possible to reduce the computation times will be explored. Future work will focus on robustness analysis and experimental validation of the proposed framework.

REFERENCES

- Aarnoudse, L., Pannier, C., Afkhami, Z., Oomen, T., and Barton, K. (2019). Multi-layer spatial iterative learning control for micro-additive manufacturing. *IFAC-PapersOnLine*, 52(15), 97–102.
- Afkhami, Z., Hoelzle, D.J., and Barton, K.L. (2021). Higher-order spatial iterative learning control for additive manufacturing. In *60th IEEE Conference on Decision and Control*, accepted. IEEE.
- Afkhami, Z., Hoelzle, D.J., and Barton, K.L. (2022). Robust higher-order spatial iterative learning control for

additive manufacturing systems. *IEEE Trans. Control Syst. Technol.*, under review.

- Afkhami, Z., Iezzi, B., Hoelzle, D., Shtein, M., and Barton, K. (2020a). Electrohydrodynamic jet printing of one-dimensional photonic crystals: Part i—an empirical model for multi-material multi-layer fabrication. *Advanced Materials Technologies*, 5(10), 2000386.
- Afkhami, Z., Pannier, C., Aarnoudse, L., Hoelzle, D., and Barton, K. (2020b). Spatial iterative learning control for multi-material three-dimensional structures. *ASME Letters in Dynamic Systems and Control*, 1(1).
- Ahn, H.S., Moore, K.L., and Chen, Y. (2007). *Iterative learning control: robustness and monotonic convergence for interval systems*. Springer Science & Business Media.
- Altin, B., Wang, Z., Hoelzle, D.J., and Barton, K. (2018). Robust monotonically convergent spatial iterative learning control: interval systems analysis via discrete Fourier transform.
- Balta, E.C., Tilbury, D.M., and Barton, K. (2019). Control-oriented modeling and layer-to-layer stability for fused deposition modeling: a kernel basis approach. In *2019 American Control Conference (ACC)*, 4727–4733. IEEE.
- Bichi, M., Ripaccioli, G., Di Cairano, S., Bernardini, D., Bemporad, A., and Kolmanovsky, I.V. (2010). Stochastic model predictive control with driver behavior learning for improved powertrain control. In *49th IEEE conference on decision and control (CDC)*, 6077–6082. IEEE.
- Bristow, D.A., Tharayil, M., and Alleyne, A.G. (2006). A survey of iterative learning control. *IEEE control systems magazine*, 26(3), 96–114.
- Gegel, M.L., Bristow, D.A., and Landers, R.G. (2019). Model predictive height control for direct energy deposition. In *Dynamic Systems and Control Conference*, volume 59148, V001T10A006. American Society of Mechanical Engineers.
- Hoelzle, D.J. and Barton, K.L. (2014). A new spatial iterative learning control approach for improved micro-additive manufacturing. In *2014 American Control Conference*, 1805–1810. IEEE.
- Hoelzle, D.J. and Barton, K.L. (2016). On Spatial Iterative Learning Control via 2-D Convolution: Stability Analysis and Computational Efficiency. *IEEE Trans. Control Syst. Technol.*
- Landers, R.G., Barton, K., Devasia, S., Kurfess, T., Pagilla, P., and Tomizuka, M. (2020). A review of manufacturing process control. *Journal of Manufacturing Science and Engineering*, 142(11). 110814.
- Lee, K.S., Chin, I.S., Lee, H.J., and Lee, J.H. (1999). Model predictive control technique combined with iterative learning for batch processes. *AIChE Journal*, 45(10), 2175–2187.
- Lequime, M., Abel-Tiberini, L., Mathieu, K., Berthon, J., and Lumeau, J. (2015). 2×2 -array pixelated optical interference filters. In *Optical Systems Design 2015: Advances in Optical Thin Films V*, volume 9627, 96270V. International Society for Optics and Photonics.
- Li, D., He, S., Xi, Y., Liu, T., Gao, F., Wang, Y., and Lu, J. (2019). Synthesis of ILC-MPC controller with data-driven approach for constrained batch processes. *IEEE Transactions on Industrial Electronics*, 67(4), 3116–3125.

- Meng, D. and Moore, K.L. (2017). Robust iterative learning control for nonrepetitive uncertain systems. *IEEE Trans. Automat. Contr.*
- Mohammadi, K., Movahhedy, M.R., and Khodaygan, S. (2019). A multiphysics model for analysis of droplet formation in electrohydrodynamic 3d printing process. *Journal of Aerosol Science*, 135, 72–85.
- Norrlöf, M. and Gunnarsson, S. (2002). Time and frequency domain convergence properties in iterative learning control. *Int. J. Control*.
- Pannek, L.G.J. and Grüne, L. (2011). Nonlinear model predictive control: Theory and algorithms. In *Nonlinear Model Predictive Control*, 2267–2274. Springer.
- Pannier, C., Wu, M., Hoelzle, D., and Barton, K. (2019). LPV models for jet-printed heightmap control. In *2019 Amer. Control Conf. IEEE*, Philadelphia, PA.
- Pannier, C.P., Ojeda, L., Wang, Z., Hoelzle, D., and Barton, K. (2018). An electrohydrodynamic jet printer with integrated metrology. *Mechatronics*, 56, 268–276.
- Rawlings, J.B., Mayne, D.Q., and Diehl, M. (2017). *Model predictive control: theory, computation, and design*, volume 2. Nob Hill Publishing Madison, WI.
- Rose, M., Barton, K., Dasgupta, N., Ransohoff, L., Herman, E., Trejo, O., Huber, C., Cho, T.H., Kazyak, E., Pannier, C.P., et al. (2021). Integrated electrohydrodynamic jet printing and spatial atomic layer deposition system for area selective-atomic layer deposition. US Patent App. 16/937,496.
- Rosolia, U. and Borrelli, F. (2017). Learning model predictive control for iterative tasks. a data-driven control framework. *IEEE Transactions on Automatic Control*, 63(7), 1883–1896.
- Spiegel, I.A., Sammons, P., and Barton, K. (2019). Hybrid modeling of electrohydrodynamic jet printing. *IEEE Transactions on Control Systems Technology*, 28(6), 2322–2335.
- Sun, J., Park, H., Kolmanovsky, I., and Choroszuca, R. (2013). Adaptive model predictive control in the ipa-sqp framework. In *52nd IEEE Conference on Decision and Control*, 5565–5570. IEEE.
- Vallon, C. and Borrelli, F. (2020). Task decomposition for iterative learning model predictive control. In *2020 American Control Conference (ACC)*, 2024–2029. IEEE.
- Wang, L., Freeman, C.T., and Rogers, E. (2016a). Predictive iterative learning control with experimental validation. *Control Engineering Practice*, 53, 24–34.
- Wang, Z., Pannier, C., Ojeda, L., Barton, K., and Hoelzle, D.J. (2016b). An application of spatial iterative learning control to micro-additive manufacturing. In *2016 American Control Conference (ACC)*, 354–359. IEEE.
- Xie, S. and Ren, J. (2018). Iterative learning-based model predictive control for precise trajectory tracking of piezo nanopositioning stage. In *2018 Annual American Control Conference (ACC)*, 2922–2927. IEEE.
- Yu, Q. and Hou, Z. (2016). Data-driven predictive iterative learning control for a class of multiple-input and multiple-output nonlinear systems. *Transactions of the Institute of Measurement and Control*, 38(3), 266–281.

APPENDIX

In this section, we describe step by step derivation of the analytical solution of the SILC-MPC framework presented

in equations (10-12). Solving (8) and (9) requires knowledge of future error signals, $\vec{e}_{l+j}$. Although we do not know these values, we can estimate them based on the values that are known from previous layer depositions,

$$\vec{e}_{l+j} = \vec{e}_l + \sum_{n=1}^j (\Delta \vec{e}_{l+n}) = \vec{e}_l + j\vec{r} - \sum_{n=1}^j \mathbf{H}_{l+n-1} \vec{f}_{l+n}. \quad (13)$$

For a single material deposition, we assume the desired height is constant for all layers such that $\Delta \vec{g}_l^d = \Delta \vec{g}_{l+n}^d = \vec{r}$. In addition, the plant spatial dynamics are noncausal in the spatial domain, meaning that the applied input at a given location will affect the output in the advanced layers and surrounding coordinates (Hoelzle and Barton (2016); Pannier et al. (2018)). This means that the error signals of future layers are functions of the input signal from prior layers, $\frac{\vec{e}_{l+i}}{\vec{f}_{l+i}} \neq 0, \forall i \leq j$. Note, the error signals of previous layers are not function of future layer inputs, $\frac{\vec{e}_{l+i}}{\vec{f}_{l+i}} = 0, \forall i > j$. From (13) we have

$$\frac{\partial \vec{e}_{l+j}}{\partial \vec{f}_{l+i}} = \mathbf{0}, \quad j < i \quad (14a)$$

$$\frac{\partial \vec{e}_{l+j}}{\partial \vec{f}_{l+i}} = -\mathbf{H}_{l+i-1}, \quad j \geq i \quad (14b)$$

To determine $\mathbf{X}$, $\mathbf{Z}$ and $\vec{Y}$ from (9), we need to calculate the partial derivative of the cost function in (8) with respect to the future layer input signals, $\frac{\partial \mathcal{J}}{\partial \vec{f}_{l+i}}$. For $i = 1$, the situation is different, $\frac{\partial \mathcal{J}}{\partial \vec{f}_{l+1}} = \frac{\partial \mathcal{J}^{mpc}}{\partial \vec{f}_{l+1}} + \frac{\partial \mathcal{J}^{ILC}}{\partial \vec{f}_{l+1}}$. We already calculated $\frac{\partial \mathcal{J}^{ILC}}{\partial \vec{f}_{l+1}}$ in (5), now we need to calculate $\frac{\partial \mathcal{J}^{mpc}}{\partial \vec{f}_{l+1}}$ as follows

$$\begin{aligned} \frac{\partial \mathcal{J}^{mpc}}{\partial \vec{f}_{l+1}} = & -\mathbf{H}_l^T \sum_{n=2}^{N_{mpc}} \mathbf{Q}_n (\vec{e}_l + n\vec{r} - \sum_{k=1}^{N_{mpc}} \mathbf{H}_{l+k-1} \vec{f}_{l+k}) - \\ & r_2 (\vec{f}_{l+2} - \vec{f}_{l+1}) - \mathbf{H}_l^T \mathbf{P}_N (\vec{e}_l + N_{mpc} \vec{r} - \sum_{k=1}^{N_{mpc}} \mathbf{H}_{l+k-1} \vec{f}_{l+k}). \end{aligned} \quad (15)$$

For $i > 1$, $\frac{\partial \mathcal{J}}{\partial \vec{f}_{l+i}} = \frac{\partial \mathcal{J}^{mpc}}{\partial \vec{f}_{l+i}} = -\mathbf{H}_{l+i-1}^T \mathbf{P}_N (\vec{e}_l + N_{mpc} \vec{r} - \sum_{n=1}^{N_{mpc}} \mathbf{H}_{l+n-1} \vec{f}_{l+n}) + \sum_{j=2}^{N_{mpc}} \frac{\partial \mathcal{J}_{l+j}}{\partial \vec{f}_{l+i}}$. Depending on the values of i and j , $\frac{\partial \mathcal{J}_{l+j}}{\partial \vec{f}_{l+i}}$ is as follows

$$\begin{aligned} i = j : & = -q_j \mathbf{H}_{l+i-1}^T (\vec{e}_l + j\vec{r} - \sum_{n=1}^{N_{mpc}} \mathbf{H}_{l+n-1} \vec{f}_{l+n}) + s_j \vec{f}_{l+j} \\ & + r_j (\vec{f}_{l+j} - \vec{f}_{l+j-1}) \end{aligned} \quad (16a)$$

$$\begin{aligned} j = i + 1 : & = -q_j \mathbf{H}_{l+i-1}^T (\vec{e}_l + j\vec{r} - \sum_{n=1}^{N_{mpc}} \mathbf{H}_{l+n-1} \vec{f}_{l+n}) \\ & - r_j (\vec{f}_{l+j} - \vec{f}_{l+i}) \end{aligned} \quad (16b)$$

$$j > i + 1 : = -q_j \mathbf{H}_{l+i-1}^T (\vec{e}_l + j\vec{r} - \sum_{n=1}^{N_{mpc}} \mathbf{H}_{l+n-1} \vec{f}_{l+n}). \quad (16c)$$

and zero for $i > j$. Summing up (16) over $j = 2, \dots, N_{mpc}$ and adding up the terminal cost components, and considering (5) and (15), $\mathbf{X}$, $\mathbf{Z}$ and $\vec{Y}$ are calculated based on the nominal plant as presented in (10-12).