

Constrained Run-to-Run Control for Precision Serial Sectioning

Damian Gallegos-Patterson^{†‡}, Kendric R. Ortiz^{†‡}, Jonathan Madison[‡], Andrew T. Polonsky[‡] Claus Danielson[†]

Abstract—This paper presents a run-to-run (R2R) controller for mechanical serial sectioning (MSS). MSS is a destructive material analysis process which repeatedly removes a thin layer of material and images the exposed surface. The images are then used to gain insight into the material properties and often to construct a 3-dimensional reconstruction of the material sample. Currently, an experience human operator selects the parameters of the MSS to achieve the desired thickness. The proposed R2R controller will automate this process while improving the precision of the material removal. The proposed R2R controller solves an optimization problem designed to minimize the variance of the material removal subject to achieving the expected target removal. This optimization problem was embedded in an R2R framework to provide iterative feedback for disturbance rejection and convergence to the target removal amount. Since an analytic model of the MSS system is unavailable, we adopted a data-driven approach to synthesize our R2R controller from historical data. The proposed R2R controller is demonstrated through simulations. Future work will empirically demonstrate the proposed R2R through experiments with a real MSS system.

I. INTRODUCTION

Serial sectioning is a destructive analysis process used to gain insight on the material properties and micro-structure of materials. Serial sectioning is typically used when non-destructive analysis is insufficient, due either to material composition or if a higher resolution of data for a specific features is required. Serial sectioning repeatedly removes thin layers of material and captures a detailed image of the exposed surface. At the end of a serial sectioning experiment, this sequence of images are combined to create a 3D reconstruction of the sample. The 3D reconstruction can be used to gain knowledge of the size, shape, and location of features of interest within the overall sample. This process is used for failure analysis, feature identification, and material composition experiments [1]. A common use is identification of flaws in 3D printed metal samples to locate the size and location of areas in the sample which leftover powered metal may still exist or where the print may have left a large void. There are three different types of serial sectioning tools, micro-mill serial sectioning, FIB laser

serial sectioning, and mechanical serial sectioning (MSS). This paper will be focusing on a MSS process.

MSS uses a three phase repetitive process of grinding, polishing, and optical imaging, to collect data from a material sample. Each cycle of grinding, polishing, and optical imaging is called a *slice* which produces a montage of images which can be stitched together to produce a larger cross-sectional image of the entire surface or a specific region of interest. Currently, an experienced human operator inputs the desired number of slices which is called a *run*. A run consists of the human operator selecting a *recipe*; a sequence of grinding pads and polishing pads, as well as the polish time, speed, solution, and solution dispensing time. The parameters of this recipe are the inputs to the MSS system. This paper will develop an autonomous controller for iteratively selecting the appropriate recipe to achieve a target material removal amount. The current method for achieving the target removal per slice is for an experienced human operator to run a series of test slices. The material removal is then measured using the average focal height of the microscope used for the image montage. Based on the calculated material removal and their experience, the human operator adjusts the recipe, typically by adjusting the polishing times for each pad. This process is repeated with another run of test slices until the target removal amount is achieved. This process is not ideal for several reasons. First, it requires significant human intervention from a skilled operator whose valuable experience could be better used for other pursuits. Second, the ‘calibration-phase’ of the recipe can require multiple test runs, removing a large amount of material. This is inappropriate for small material-samples since a significant portion of the material-sample will not be sectioned at a consistent rate. Third, pad wear over the course of a long run can cause the removal amount to drop causing inconsistent slice thickness during image reconstruction. An automated MSS controller could both reduce human intervention and improve the performance of the system.

The contributions of this paper are summarized as follows. This paper proposes a run-to-run (R2R) controller to reduce operator intervention while improving the precision and consistency of material removal. Run-to-run (R2R) is described as “similar to batch processes but more extensive” [2]. R2R is commonly used to reduce output variance and increase precision in systems that perform repetitive processes [3]–[8]. R2R assumes that the output is sparsely sampled which allows for a linear regression model to be used. We will be using R2R in combination with constrained optimization

This material is based upon work supported by the National Science Foundation under NSF Grant Number CNS-2105631. 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. Sandia National Laboratories is a multimission laboratory managed and operated by National Technology and Engineering Solutions of Sandia, LLC., a wholly owned subsidiary of Honeywell International, Inc., for the U.S. Department of Energy’s National Nuclear Security Administration under contract DE-NA-0003525. [†]University of New Mexico. [‡]Sandia National Laboratories. Corresponding Author: Damian Gallegos-Patterson dagallegospatterson@unm.edu

with the goal of reducing the variance of the system output while achieving the target removal amount faster and more consistently than the “guess and check” current method being used. One of the main challenges is that we do not have access to an analytic first-principles model that maps recipes to material removal. Instead, we will use a data-driven method to synthesize our R2R controller from historical operational data. We will also compare our method with a state of the art R2R method.

Notation: The mean and variance of a random variable x will be denoted by $\mathbb{E}[x]$ and $\mathbb{V}[x] = \mathbb{E}[xx^\top] - \mathbb{E}[x]\mathbb{E}[x]^\top$. The Kronecker product of matrices A and B is denoted by $A \otimes B$. The vectorization $\text{vec}(A) \in \mathbb{R}^{nm}$ of a matrix $A \in \mathbb{R}^{n \times m}$ is a vector comprised of the concatenated column vectors of A . For a matrix A with linearly independent columns, its left pseudo-inverse is a matrix A^+ such that $A^+A = I$. In particular, the Moore-Penrose pseudo-inverse is $A^+ = (A^\top A)^{-1}A^\top$.

II. AUTOMATING MSS PROBLEM

In this section, we begin with a brief description of the MSS plant, followed by the operational constraints, and the control objectives. We then provide a description of the historical data available to us, which will be used for data-driven controller synthesis.

A. The MSS Plant

We begin by considering a deterministic system model for the MSS plant. Due to the operational nature of MSS systems, we do not have direct access to in-situ measurements, nor can we change the recipe during a run. Thus, we model the MSS system plant as a discrete time algebraic map of the recipe u_i to removal amount y_i

$$y_i = f(u_i, d_i) \quad (1)$$

where $u_i \in \mathbb{R}^{n_u}$ and $y_i \in \mathbb{R}^1$ are the MSS i^{th} system inputs and outputs respectively, and the ‘time’-index i represents the *slice number* for the plant. We consider the plant (1) to be deterministic, but unknown. We note that the plant (1) ‘dynamics’ do not depend on the system ‘state’ y_i . Thus, the plant is a static non-linearity mapping $f(u_i, d_i) : \mathbb{R}^{n_u} \times \mathbb{R}^{n_d} \rightarrow \mathbb{R}^1$ that maps recipes u_i and additional hidden variables d_i to the removal amount y_i . The hidden variables $d_i \in \mathbb{R}^{n_d}$ characterize all unknown factors within the system, such as material hardness, thermal effects, sensor measurement error, grinding and polishing pad wear, cavities in the material, etc. Since we do not have access to the hidden variables d_i , it is not possible to learn a model of the static nonlinearity (1) from historical data. Furthermore, even if the plant model were known, it cannot be utilized since the hidden variables d_i are unmeasured and time-varying. Instead, we will use real-time feedback to adjust the recipe u_i online to achieve the desired removal amount $y_i \rightarrow r$ where $r \in \mathbb{R}^1$ is the target removal amount. Feedback control is ideally suited to the problem of rejecting unknown and varying disturbances d_i . Although the plant (1)

is static, the closed-loop system will be dynamic due to the dynamics of the controller.

The MSS plant (1) is an over-actuated system; there are multiple $n_u > n_y = 1$ inputs $u_i \in \mathbb{R}^{n_u}$ that can be manipulated to drive $y_i \rightarrow r$ one output y_i to the desired removal amount $r \in \mathbb{R}^1$. The manipulated inputs are the recipe parameters summarized in Table I. For the automated system, we envision that the MSS will operate with a fixed sequence of multiple grinding and polishing pad types, as well as different solution types for each pad. The automated system will then select the polishing speed and polishing time for each of the pads in the sequence. The vector $u_i \in \mathbb{R}^{n_u}$ of polishing-speeds and polishing-times is the control input (recipe) for the MSS system. For this preliminary work, we will only manipulate the polishing times for one grinding pad followed by one polishing pad. Thus, we have a 2 input system. Importantly for controller development, the removal amount (1) is monotonically non-decreasing with respect to the polish times i.e. polishing for a longer time will not result in less material removed.

The over-actuation of the MSS renders human-in-the-loop operation difficult, requiring the operator to have significant experience and expertise to choose the appropriate recipe $u \in \mathbb{R}^{n_u}$ to achieve the desired removal amount $y_i \in \mathbb{R}^1$. An automated system will improve the user-friendliness of the MSS. Furthermore, an automated system could harness this over-actuation to improving system performance, shorten the calibration-phase, and reduce human error. Performance objectives will be described in Section II-C.

TABLE I
INPUTS (RECIPE PARAMETERS) FOR THE MSS

Input Variable	Constraints
u_{rt} cutting pad type	$1 \leq \gamma \leq 8$
u_{pt} polishing pad type	$1 \leq \gamma \leq 8$
u_s polishing solution type	$1 \leq \gamma \leq 4$
u_ω polishing speed (RPM) for each pad	$1 \leq \gamma \leq 300$
u polishing time for each pad	$5 \leq \gamma$

B. Operational Constraints

Our controller must produce recipes u that are physically implementable by the MSS. The operational constraints shown in Table I describe the physical limitations of the MSS system. The maximum number of pads the system is able to hold at a time is eight, therefore $u_{rt} + u_{pt} \leq 8$ for any recipe. For polishing solutions u_s we are able to select one one of the following particle sizes $1\mu\text{m}$, $3\mu\text{m}$, $6\mu\text{m}$, or $9\mu\text{m}$ which can be used in any combination with the pad types. The speed at which the pads can be rotated u_ω is limited to 300 RPM. The minimum amount of time a pad can polish for is five seconds. To ensuring a favorable imaging surface while avoiding over use of polishing pads, constraining polishing pads to more than sixty seconds and less than 200 seconds will be imposed. The constraints for

our system form a n_u -dimensional polytope

$$\mathcal{U} = \{u \in \mathbb{R}^{n_u} : Hu \leq h\}. \quad (2)$$

We will analyze our R2R controller with and without these constraints (2).

C. Control Objectives

Beyond automating the operation of the MSS, our control objectives include improving its performance. Our control objectives can be summarized by the following conceptual stochastic optimization problem

$$u_i = \arg \min_u \mathbb{V}[y - r|u] \quad (3a)$$

$$\text{s.t. } \mathbb{E}[y|u] = r \quad (3b)$$

$$u \in \mathcal{U}. \quad (3c)$$

The desired controller should compute a recipe u such that the expected material removal $\mathbb{E}[y|u]$ matches the target removal amount r i.e. the recipe u should satisfy the equality constraint (3b). Since the MSS is over-actuated $n_u > 1$, there are potentially an infinite-number of recipes $u \in \mathbb{R}^{n_u}$ that can achieve the desired removal (3b). Among these recipes u , we would like to select the recipe that produces the lowest variance (3a) so that the slices have uniform thickness. Finally, the recipe u must be implementable (3c) given the input constraints described in Section II-B.

In Section III, we will translate the conceptual stochastic optimization problem (3) into an implementable deterministic optimization problem. We will use a data-driven approach to formulate this deterministic optimization problem from historical operation data. This deterministic optimization problem will be embedded in a R2R control framework to iteratively ensure that the removal amount converges $y_i \rightarrow r$ to the target removal amount r .

D. Historical Operational-Data

We will use a data-driven approach to translate the conceptual stochastic optimization problem (3) into an implementable deterministic form. The available data consists of pairs of recipes $u_i \in \mathbb{R}^{n_u}$ and the resulting removal rate $y_i \in \mathbb{R}^1$. The entire available data set $\mathcal{D} = \{\mathcal{Y}, \mathcal{U}\}$ is comprised of historical usage of the plant providing a set of outputs based off of previous operator defined inputs. The set $\mathcal{D} \in \mathbb{R}^{N \times n_D}$ is the set of N states corresponding to the set of inputs $\mathcal{U} \in \mathbb{R}^{N-1 \times n_u}$ with respective outputs $\mathcal{Y} \in \mathbb{R}^{N \times n_y}$. Using this historical data, we are able to approximate a deterministic model for (1).

III. OPTIMAL RUN-TO-RUN CONTROLLER

In this section, we describe the proposed R2R controller for automating the MSS system. R2R control is the appropriate paradigm for this problem due to the lack of in-situ measurements and our inability to alter the recipe during a slice. Our algorithm embeds a deterministic formulation of the stochastic optimization problem (3) into a R2R framework in order to compute optimal recipes u_{i+1}^* . The R2R framework provides a feedback mechanism for adjusting the

recipe u_{i+1}^* based on the material removal y_i , which is measured after each slice. This feedback is used to reject the hidden variables d_i , which we consider disturbances.

A. Optimal Recipe

The main challenge for MSS controller synthesis is that the plant model (1) mapping recipes u_i to removal amount y_i is an *unknown* static non-linearity. However, since the removal amount (1) is monotonic, we can use a linear approximation

$$y_i = c + b^\top u_i \quad (4)$$

where $c \approx f(u_i, d_i)$ is the drift coefficient and $b \approx \nabla f(u_i, d_i)$ is the slope coefficient around the operating point u_i . The parameters $c \in \mathbb{R}^1$ and $b \in \mathbb{R}^{n_u}$ are uncertain and time-varying due to both the changing linearization point u_i and the hidden variables d_i . To capture this uncertainty, we will model these parameters as stochastic. Although the probability density function of these stochastic variables is unknown, we will use historical-data to quantify our uncertainty using their empirical moments. We will use these empirical moments to translate the conceptual stochastic optimization problem (3) into a deterministic optimization problem. Since we consider stochastic *parameters*, estimating the parameters is non-trivial.

First, we translate the stochastic equality constraint (3b) into a deterministic constraint based on empirical moments. Substituting the stochastic linear model (4) into the equality constraint (3b) yields $\mathbb{E}[y_i|u_i] = \mathbb{E}[c + b^\top u_i|u_i] = r_i$. Exploiting the linearity of the expectation, we obtain $\mathbb{E}[c] + \mathbb{E}[b]^\top u_i = r_i$ where r_i and u_i are deterministic. This equality becomes the deterministic constraints (6b) when the expectations $\mathbb{E}[c]$ and $\mathbb{E}[b]$ are replaced by their empirical estimates $\mu_c \approx \mathbb{E}[c]$ and $\mu_b \approx \mathbb{E}[b]$.

Next, we translate the conceptual stochastic cost (3a) into a deterministic cost function. Substituting the stochastic linear model (4) into the cost (3a) yields

$$\mathbb{V}[y_i|u_i] = \mathbb{E}[(y_i - r_i)^2|u_i] = \mathbb{E}[(c + b^\top u_i - r_i)^2|u_i] \quad (5)$$

where the mean value of y_i is r_i due to the equality constraint (3b). Expanding the cost, yields $\mathbb{V}[y_i|u_i] \propto u_i^\top \mathbb{E}[bb^\top] u_i + 2u_i^\top \mathbb{E}[bc] - 2u_i^\top \mathbb{E}[b] r_i$ where the terms from $\mathbb{E}[(c + r_i)^2]$ were omitted since they do not depend on the decision variables u_i . Substituting the empirical estimates $\mathbb{E}[bb^\top] \approx \Sigma_{bb} + \mu_b \mu_b^\top$ and $\mathbb{E}[bc] \approx \Sigma_{bc} + \mu_b \mu_c$, we obtain $\mathbb{V}[y_i|u_i] \propto u_i^\top (\Sigma_{bb} + \mu_b \mu_b^\top) u_i + 2u_i^\top \Sigma_{bc} - 2u_i^\top \mu_b (\mu_c - r_i)$. Since $\mu_c - r_i = -\mu_b^\top u_i$ according to (6b), we obtain $\mathbb{V}[y_i|u_i] \propto u_i^\top (\Sigma_{bb} - \mu_b \mu_b^\top) u_i + 2u_i^\top \Sigma_{bc}$. Finally, noting that $\mu_b^\top u_i$ is constant, we obtain the deterministic cost (6a). Thus, the conceptual stochastic optimization problem (3) can now be approximated by the following deterministic optimization problem

$$u_i = \arg \min_u u^\top \Sigma_{bb} u + 2u^\top \Sigma_{bc} \quad (6a)$$

$$\text{s.t. } \mu_c + \mu_b^\top u_i = r_i \quad (6b)$$

$$u_i \in \mathcal{U} \quad (6c)$$

where the approximation is due to the use of empirical

estimates of the means μ_c , μ_b and variances Σ_{bb} , Σ_{bc} of the parameters. Conveniently, this problem formulation (6) only requires second-order statics for the model (4) parameters. Solving (6) will produce the optimal recipe u_i^* .

B. Run-to-Run Controller

The deterministic optimization problem (6) is static. Thus in this section, we embed this optimization problem (6) into an R2R framework to provide feedback. Our R2R algorithm iteratively adjusts the recipe u_{i+1}^* based on the measured material removed y_i during the previous slice i . The R2R feedback allows the material removal to converge $y_i \rightarrow r$ to the desired removal amount r while rejecting the unmeasured disturbances d_i . The difference $y_i - r$ between the actual y_i and desired r removal amount is used to update the drift coefficient μ_c . We update the drift coefficient using exponentially weighted moving average (EWMA) dynamics [9]

$$\mu_{c,i+1} = \mu_{c,i} + \lambda(y_i - r) \quad (7)$$

where $\mu_{c,i+1}$ is the updated drift coefficient and $\lambda \in [0, 1]$ is a tuning parameter. The EWMA update-law has many beneficial properties [9]. The optimization problem (6) is solved with the updated drift coefficient μ_c to obtain a new recipe u_i which is then implemented. Thus, we can interpret the drift coefficient μ_c as a state and the equality constraint (6b) as dynamics. The R2R controller continues to refine the recipe u_i until all slices have been completed.

C. Comparison with Existing R2R Controllers

In this section, we compare our R2R controller algorithm with existing R2R controllers from the literature. As noted in the survey [2], most R2R controllers have the following integral dynamics

$$u_{i+1} = u_i + \lambda \mu_b^+ (r - y_i) \quad (8)$$

where $\mu_b^+ = \mu_b / \mu_b^\top \mu_b$ is the Moore-Penrose pseudo-inverse and $\lambda \in [0, 1]$ is a tuning parameter. See equation (16) from [2] for details. We will show that our R2R controller has integral dynamics (8) when the input constraints (6c) are ignored, although with a novel pseudo-inverse. In contrast, when the input constraints are included the integral dynamics no longer apply. Without the input constraints (6c), the Karush-Kuhn-Tucker (KKT) optimality conditions for (6) are

$$\begin{bmatrix} 2\Sigma_{bb} & \mu_b \\ \mu_b^\top & 0 \end{bmatrix} \begin{bmatrix} u \\ \nu \end{bmatrix} = \begin{bmatrix} -2\Sigma_{bc} \\ r - \mu_c \end{bmatrix} \quad (9)$$

where $\nu \in \mathbb{R}$ is the dual variable associated with the equality constraint (6b). Solving (9) for u , we obtain the control-law

$$u = \mu_b^+ (r - \mu_c) + \frac{1}{2} (I - \mu_b^+ \mu_b^\top) \Sigma_{bb}^{-1} \Sigma_{bc} \quad (10)$$

where $\mu_b^+ = \Sigma_{bb}^{-1} \mu_b / (\mu_b^\top \Sigma_{bb}^{-1} \mu_b)$ is an alternative pseudo-inverse of μ_b i.e. $\mu_b^\top \mu_b^+ = \mu_b^\top \Sigma_{bb}^{-1} \mu_b / (\mu_b^\top \Sigma_{bb}^{-1} \mu_b) = 1$. This pseudo-inverse was derived from (6) to minimize the variance $\mathbb{E}[(y - r)^2 | u]$ of the material removal. Combining (10) with the EWMA dynamics (7), we obtain the following

integral dynamics

$$\begin{aligned} u_{i+1} &= \mu_b^+ (r - \mu_{c,i}) + \frac{1}{2} (I - \mu_b^+ \mu_b^\top) \Sigma_{bb}^{-1} \Sigma_{bc} + \lambda \mu_b^+ (r - y_i) \\ &= u_i + \lambda \mu_b^+ (r - y_i) \end{aligned} \quad (11)$$

with the specific initial condition $u_0 = \mu_b^+ (r - \mu_{c0}) + \frac{1}{2} (I - \mu_b^+ \mu_b^\top) \Sigma_{bb}^{-1} \Sigma_{bc}$. This initial condition is important since otherwise the dynamic controller (11) would not include the second-term which compensates for possible cross-correlation between the model (4) parameters c and b . Note that our integral dynamics (11) match the literature dynamics (8).

Next, we show our R2R Algorithm does not necessarily have literature dynamics (8) when the input constraints (6c) are included. With the input constraints (6c), the KKT optimality conditions for (6) are

$$\begin{bmatrix} 2\Sigma_{bb} & \mu_b & H_A^\top \\ \mu_b^\top & 0 & 0 \\ H_A & 0 & 0 \end{bmatrix} \begin{bmatrix} u \\ \nu \\ \lambda_A \end{bmatrix} = \begin{bmatrix} -2\Sigma_{bc} \\ r - \mu_c \\ h_A \end{bmatrix} \quad (12)$$

where H and h are the half-space parameters of the input constraints (2) and H_A and h_A are the rows corresponding to the subset $\mathcal{A}$ of constraints that are active at the optimal. The active dual variables are denoted by $\lambda_A \geq 0$ where the dual variable corresponding to inactive constraints are zero. Through brute-force computation, we obtain

$$u = \mu_b^+ (r - \mu_c) + \frac{1}{2} (I - \mu_b^+ \mu_b^\top) (\Sigma_{bb}^{-1} \Sigma_{bc} + \Gamma h_A) \quad (13)$$

where $\Gamma = \Sigma_{bb}^{-1} H_A^\top (H_A \Sigma_{bb}^{-1} H_A^\top)^{-1}$ and μ_b^+ is yet another pseudo-inverse of μ_b given by

$$\mu_b^+ = \frac{(\Sigma_{bb}^{-1} - \Sigma_{bb}^{-1} H_A^\top (H_A \Sigma_{bb}^{-1} H_A^\top)^{-1} H_A \Sigma_{bb}^{-1}) \mu_b}{\mu_b^\top (\Sigma_{bb}^{-1} - \Sigma_{bb}^{-1} H_A^\top (H_A \Sigma_{bb}^{-1} H_A^\top)^{-1} H_A \Sigma_{bb}^{-1}) \mu_b}.$$

Note that if (6) is feasible then μ_b does not lie in the null-space of $\Sigma_{bb}^{-1} - \Sigma_{bb}^{-1} H_A^\top (H_A \Sigma_{bb}^{-1} H_A^\top)^{-1} H_A \Sigma_{bb}^{-1}$. This follows from the fact that the active inequality constraints cannot bind the equality constraint (6b).

Although (13) has a similar structure as (10), it cannot necessarily be transformed into the integral-form (8). As μ_c changes (7), the optimal active-set $\mathcal{A}$ can change. Thus, the pseudo-inverse μ_b^+ and matrix Γ are time-varying. Thus, the nonlinear map provided by the optimization problem (6) replaces rather than integrates (8) the control inputs. Note that, although our R2R controller does not have the integral dynamics (8), it is still dynamic due to the EWMA dynamics (7). Finally, note that our R2R controller can be trivially put in the general form $u_{i+1} = \alpha u_i + \delta u_i$ given by equation (17) in [2] since any arbitrary feedback controller $\kappa(x)$ can be written in this form by defining $\delta u_i = \kappa(x) - \alpha u_i$.

IV. NUMERICAL RESULTS

In this section, we present numerical results demonstrating our R2R controller. First, we present results verifying our stochastic model (4) of the material removal (1). Next, we present simulation results that demonstrate our R2R controller for an unknown static nonlinearity (1). Finally, we present simulations results of our R2R controller for

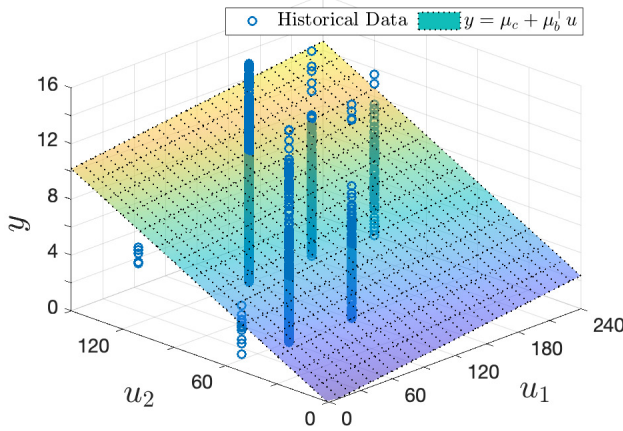


Fig. 1. Fit of stochastic linear model (4) using historical data.

the stochastic model (4) where the parameters c and b are time-varying with a Gaussian distribution. Throughout this section, we consider the MSS with 2 pads to allow us to plot $y_i \in \mathbb{R}^1$ versus $u_i \in \mathbb{R}^2$.

A. Parameter Estimation and Model Validation

We use historical operational data to estimate the mean and variance of the stochastic linear model (4). Historical data was collected over a ten years span and contain over 150 runs containing up to 500 slices per run. The data files contain the inputs of the system (polish times, RPM, pads used) and outputs (microscope focus height). We utilize the observations from the system plant which come in the form of multiple data files, which contain the inputs (polish times, RPM, pads used) and outputs (microscope focus height). We developed a script that extracts this data from thousands of separate text-files and collects the data into a unified data set $\mathcal{D} = (\mathcal{Y}, \mathcal{U})$ where $\mathcal{U} \in \mathbb{R}^{n \times 2}$ are the recipes and $\mathcal{Y} \in \mathbb{R}^{n \times 1}$ are the resulting amounts of material removed. When a slice is imaged, the focal height of the microscope is recorded for each image in the montage. The average focal height is then calculated and used to estimate the amount of material removed for each slice. Microscope auto focus errors can occur which will cause incorrect average focal height readings which can lead to readings of negative or minimal removal amounts. Therefore, the need to pre-process the data by removing outliers is required. This includes removing all data points associated with negative values. Also, we compute a preliminary estimate of the mean and variance of the parameters c and b . Any data-points $y_i = c + b^T u_i$ outside of 3 standard deviations of the estimated mean value $y_i = \mu_c + \mu_b^T u_i$ are removed. Once pre-processing is complete, we use the historical data to estimate the mean and variance of the parameters of the stochastic model (4) using the generalized method of moments (GMM) method [10]. The curve-fit of the mean $y_i = \mu_c + \mu_b^T u_i$ of (4) is shown in Fig. 1.

Fig. 1 shows that the available historical data is not very exciting (in the sense of persistency of excitation). The expert human operators tend to use a few different recipes and the

polish times are round numbers, typically multiples of 60 seconds. Indeed, the excitation of this input data u_i is

$$\underline{\sigma} \left(\frac{1}{N} \sum_{i=1}^N \frac{u_i u_i^T}{u_i^T u_i} \right) = 0.0678$$

where $\underline{\sigma}(\cdot)$ is the smallest singular-value of a matrix. While the low-level of excitation makes it difficult to accurately estimate the parameters, it demonstrates the room for improvement through automation. The proposed R2R control algorithm will not artificially restrict itself to a small number of recipes with round numbers. This greater flexibility can potentially lead to improved performance. In future work, we will consider active-learning/dual-control to produce more exciting data for our data-driven R2R control design.

Fig. 1 shows that material removal y_i is highly variable. Even when the same recipe $u_i = \bar{u}$ is used, the resulting removal y_i varies greatly. This is partially due to measurement noise, but the hidden parameters d_i play a significant role. Polishing a soft material will remove far more material than polishing a hard material for the same amount of time. Likewise, a fresh pad will remove material more quickly than an old pad. This high variance of the material removal shown in Fig. 1 motivates our decision to model the material removal as stochastic (4). This also motivates our objective of finding recipes that minimize the variance in the removal of material.

B. Simulation Results

In this section, we compare our R2R controller in with the literature controller (8) using the Moore-Penrose pseudo-inverse $\mu_b^\dagger = \mu_b / \mu_b^T \mu_b$ of μ_b . We also compare our controller without the input constraints (6c). As we showed in Section III-C, without (6c) our controller has the form (8) with the pseudo-inverse

$$\mu_b^\dagger = \frac{\Sigma_{bb}^{-1} \mu_b}{\mu_b^T \Sigma_{bb}^{-1} \mu_b}. \quad (14)$$

For these simulation results, we model the removal function (1) as $y_i = \mu_c + \mu_b^T u_i$ where y_i is the removal amount, μ_c is estimated variance of the output, μ_b is the estimated variance of the inputs, and u_i is the inputs. The simulation results are shown in Fig. 2. For each of the 3 R2R algorithms we show the removal amount y_i and the recipe u_i versus slice i . The desired removal amount is $r = 10.5$ microns.

Each of the R2R algorithms converged to the desired removal amount $y_i \rightarrow r$ after 14 slices. This is fast convergence considering an experiment is typically comprised of hundreds of slices. Furthermore, a human operator can require up to 40 slices to find an appropriate recipe for a unique sample with adjustments needed during the course of an experiment. However, the two linear R2R controllers (8) with different pseudo-inverses produced different recipes to achieve the desired removal amount.

Figure 2 and 3 shows both linear R2R controllers (8) produced non-implementable recipes. In Figure 2, the literature R2R controller (8) with the Moore-Penrose pseudo-

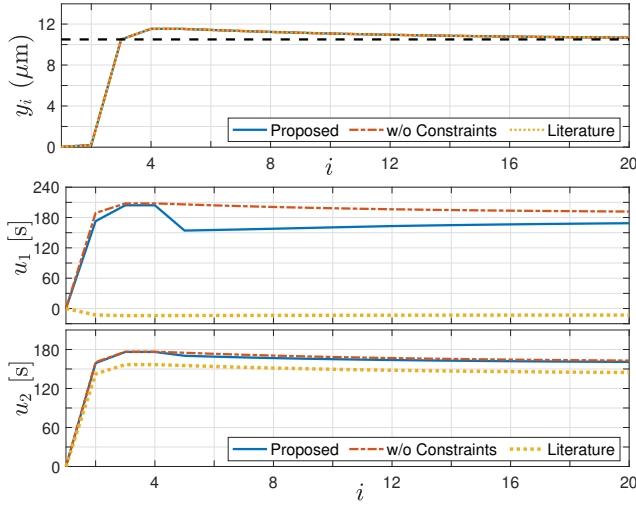


Fig. 2. Simulation results comparing the proposed R2R control with and without constraints with the literature R2R controller using the Moore-Penrose pseudo-inverse.

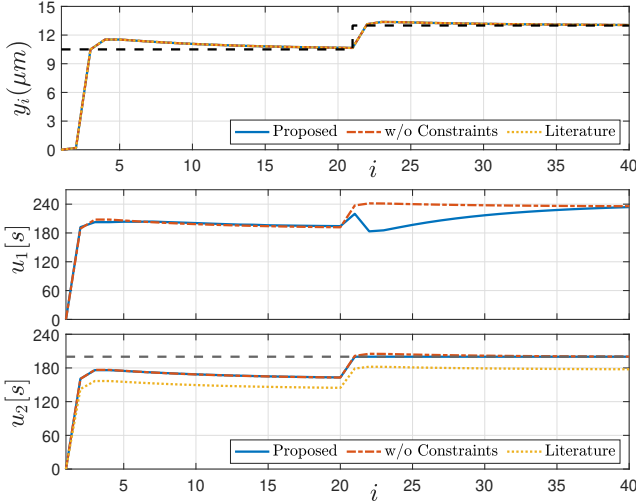


Fig. 3. Simulation results comparing the proposed R2R control with and without constraints with the literature R2R controller (8) using the Moore-Penrose pseudo-inverse.

inverse produces a negative polishing times for one of the pads, which is obviously unimplementable. Figure 3 shows that changes in the target removal amount leads to the unconstrained variant of our controller to produce an excessively long polishing times for one of the pads that violated the constraints.

C. Robustness Simulations

In this section, we present simulation results that stress-test the R2R algorithms. We modeled the material removal function (1) using the linear model (4) with time-varying parameters c and b . For each slice i , the parameters were sampled from a Gaussian distribution

$$\begin{bmatrix} c \\ b \end{bmatrix} \sim \mathcal{N} \left(\begin{bmatrix} \mu_c \\ \mu_b \end{bmatrix}, \begin{bmatrix} \Sigma_{cc} & \Sigma_{cb} \\ \Sigma_{bc} & \Sigma_{bb} \end{bmatrix} \right)$$

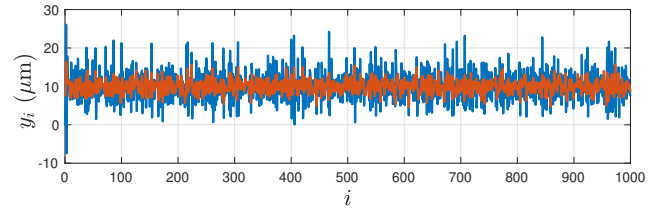


Fig. 4. Comparison of R2R controllers for stochastic removal amounts

where the mean and variance were empirically estimated. Our R2R controller was compared with the literature controller (8) using the Moore-Penrose pseudo-inverse. The simulation results are shown in Fig. 4. The target removal amount was $r = 10$ microns. For a fair comparison, the sequence of parameters c_i and b_i was pre-computed so that both algorithms had the same realizations of the random variables. As Fig. 4 shows, both R2R algorithms keep the average removal amount $\mathbb{E}[y_i]$ around the target removal amount. Indeed, our R2R algorithm had an average removal amount of $\mathbb{E}[y_i] = 9.99$ whereas the literature algorithm had an average $\mathbb{E}[y_i] = 9.98$. The advantage of our R2R controller is the reduction of the variance of the removal amount $\mathbb{V}[y_i]$. The reduction in the variance is apparent from Fig. 4. Indeed, the variance of the removal amount for our algorithm was $\mathbb{V}[y_i] = 3.33$ whereas the variance of the literature controller was $\mathbb{V}[y_i] = 17.69$. Thus, for these simulation results, our controller reduces the variance of the removal amount by 81%.

REFERENCES

- [1] J. Madison, J. Spowart, D. Rowenhorst, J. Fiedler, and T. Pollock, "Characterization of three-dimensional dendritic structures in nickel-base single crystals for investigation of defect formation," in *Proceedings of the Superalloys Conference*. Roger C. Reed and at. Champion Pennsylvania USA The Minerals, Metals . . . , 2008.
- [2] Y. Wang, F. Gao, and F. J. Doyle, "Survey on iterative learning control, repetitive control, and run-to-run control," *J of Process Control*, 2009.
- [3] A. Rzepniewski and D. E. Hardt, "Development of general multivariable run-by-run control methods with application to a sheet metal forming process," *International J of Machine Tools & Manufacture*, 2008.
- [4] R. Ganesan, T. K. Das, and K. M. Ramachandran, "A multiresolution analysis-assisted reinforcement learning approach to run-by-run control," *IEEE Transactions on Automation Science and Engineering*, 2007.
- [5] J. Busch and W. Marquardt, "Run-to-run control of membrane filtration in wastewater treatment - an experimental study 1," *IFAC Proceedings Volumes*, 2007, 8th IFAC Symposium on Dynamics and Control of Process Systems.
- [6] K. Miwa, T. Inokuchi, T. Takahashi, A. Oikawa, and K. Imaoka, "Reduction of depth variation in an si etching process by applying an optimized run-to-run control system," *IEEE Transactions on Semiconductor Manufacturing*, 2005.
- [7] E. Hamby, P. Kabamba, and P. Khargonekar, "A probabilistic approach to run-to-run control," *IEEE Transactions on Semiconductor Manufacturing*, 1998.
- [8] R.-S. Guo, A. Chen, and J.-J. Chen, "Run-to-run control schemes for cmp process subject to deterministic drifts," in *2000 Semiconductor Manufacturing Technology Workshop (Cat. No.00EX406)*, 2000.
- [9] J. Moyne, "Run-to-run control in semiconductor manufacturing," in *Encyclopedia of Systems and Control*. Springer London, 2014.
- [10] L. P. Hansen, "Large sample properties of generalized method of moments estimators," *Econometrica*, 1982.