IIE Transactions

ISSN: 0740-817X (Print) 1545-8830 (Online) Journal homepage: https://www.tandfonline.com/loi/uiie20

Reconfigured piecewise linear regression tree for multistage manufacturing process control

Ran Jin & Jianjun Shi

To cite this article: Ran Jin & Jianjun Shi (2012) Reconfigured piecewise linear regression tree for multistage manufacturing process control, IIE Transactions, 44:4, 249-261, DOI: 10.1080/0740817X.2011.564603

To link to this article: https://doi.org/10.1080/0740817X.2011.564603

Accepted author version posted online: 24 May 2011.
Published online: 23 Jan 2012.

IIE Transactions (2012) **44**, 249–261

ISSN: 0740-817X print / 1545-8830 online
DOI: 10.1080/0740817X.2011.564603

Reconfigured piecewise linear regression tree for multistage manufacturing process control

RAN JIN and JIANJUN SHI*

H. Milton Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology, Atlanta, GA 30332, USA
E-mail: Jianjun.shi@isye.gatech.edu

Received March 2010 and accepted January 2011

In a multistage manufacturing process, extensive amounts of observational data are obtained by the measurement of product quality features, process variables, and material properties. These data have temporal and spatial relationships and may have a non-linear data structure. It is a challenging task to model the variation and its propagation using these data and then use the model for feedforward control purposes. This article proposes a methodology for feedforward control that is based on a piecewise linear model. An engineering-driven reconfiguration method for piecewise linear regression trees is proposed. The model complexity is further reduced by merging the leaf nodes with the constraint of the control accuracy requirement. A case study on a multistage wafer manufacturing process is conducted to illustrate the procedure and effectiveness of the proposed method.

Keywords: Automatic process control, engineering-driven reconfiguration, manufacturing, piecewise linear regression tree

1. Introduction

A Multistage Manufacturing Process (MMP) refers to a manufacturing system that consists of multiple units, stations, or operations that are used to create a product. In most cases, the quality of the product created by the MMP is determined by complex interactions among the multiple stages. The quality characteristics of a stage are not only influenced by the local variations at that stage but also by the propagated variations from upstream stages. An MMP presents significant challenges, as well as opportunities, for quality engineering research. Two of these challenges are how to model the variation and its propagation along the production stages and how to further use the model to reduce the variation in the final product.

Various methodologies have been developed to both model and control system variability in MMPs. Feedforward control is one of the commonly adopted methodologies for such purposes. There are three main feedforward control strategies reported in the literature that can be characterized in terms of the model used to represent the MMP.

One methodology is called Stream of Variation (SOV) and is based on a state–space model (Jin and Shi, 1999; Shi, 2006). An SOV model is typically obtained from engineering knowledge, such as design information and physical laws of the process. The literature on feedforward control under the SOV framework includes papers on the adjustment of the fixture position and the tool path in a machining process (Djurdjanovic and Zhu, 2005) and variation reduction in an assembly process when taking the controllability and measurement noises into account (Izquierdo *et al.*, 2007). In recent years, a new control strategy has been developed that is based on a one-step-ahead optimal criterion. The control actions are updated iteratively as the operations progress (Jiao and Djurdjanovic, 2010). The control performance of this type of approach depends on the validity and accuracy of the state–space model. The SOV-based feedforward control approach may not be applicable if (i) an SOV model cannot be obtained in terms of physics and engineering knowledge because of the complexity of the system or (ii) there are strong non-linear relationships among process variables and quality variables. In this situation, an effective data-driven modeling method is desirable to address the non-linear properties of the observational data.

Other methodologies have been developed based on regression models, such as Robust Parameter Design (RPD)-based feedforward control (Joseph, 2003) and Design of Experiments (DOE)-based Automatic Process Control (APC; Jin and Ding, 2004). DOE-based APC determines the control actions by minimizing the predicted control objective function from a global regression model. Certainty equivalence control or cautious control strategies are used in the APC approach (Jin and Ding, 2004). Recently, Zhong *et al.* (2010) investigated the impact of model uncertainty and sensing error on the control performance. The DOE-based APC approach yielded better performance for

*Corresponding author

variability reduction than the traditional RPD approach. However, the DOE-based APC approach has two limitations: (i) the global regression model predicts the final quality variables when information at all stages is known. Thus, it cannot be used for control at an intermediate stage when only upstream stage information is available; and (ii) the single regression model strategy is unable to address complex situations in an MMP when the data structure is non-linear.

The extensive observational data that are collected in a modern MMP means that there is timely information available about the process variables, material properties, and intermediate quality measures. Thus, data-mining techniques can be applied to this data to isolate the interrelationships among the variables. Regression tree models are an effective approach to modeling non-linear data structures; their use creates a high prediction accuracy and allows explicit interpretation of predictors. Therefore, regression tree models are adopted in this article to model the variation and its propagation in MMPs.

There are three typical methods to model a regression tree: (i) greedy search; (ii) Bayesian trees; and (iii) statistical tests. In general, the greedy search approaches are biased in splitting variable selection and are computationally intensive; examples of this approach include the AID algorithm (Morgan and Sonquist, 1963) and the classification and regression tree approach (Breiman *et al.*, 1984). The Bayesian tree approach can be used to improve the computational efficiency and depends on proposing the prior distributions for both the tree structure and parameters (Chipman *et al.*, 1998, 2002; Denison *et al.*, 2002). The Markov chain Monte Carlo method is used to determine the posterior distributions. Another approach is to use statistical tests to determine splitting variables, such as smoothed and unsmoothed piecewise-polynomial regression trees (Chaudhuri *et al.*, 1994) and Generalized, Unbiased Interaction Detection and Estimation (GUIDE; Loh, 2002; Kim *et al.*, 2007). In these approaches, the residuals of piecewise models are tested with high computational efficiency.

In this article, Piecewise Linear Regression Trees (PLRTs) estimated by GUIDE are adopted to model MMPs for process control. The reasons for selecting PLRTs using GUIDE are as follows: (i) a PLRT from GUIDE has better prediction accuracy for a non-linear data structure than a global regression model (Loh, 2002; Kim *et al.*, 2007; Loh *et al.*, 2007); (ii) the interpretation of the PLRT is explicit. The predictors in the tree structure are explained as important factors under different scenarios or splitting conditions; and (iii) GUIDE has several superior properties over other estimation methods. For example, both categorical and continuous predictors can be assigned to different roles, such as splitting only, regression only, or both. It also alleviates the selection bias and investigates the local pairwise interactions. Therefore, it is an effective way to link the process, material properties, and quality variables in MMPs.

A PLRT from GUIDE performs well for quality prediction in MMPs but not for variation reduction. There are two major limitations that prohibit using a PLRT directly in feedforward control for variation reduction.

1. In an MMP, the temporal orders are determined by the design of the manufacturing system. However, the splitting order in PLRTs is prioritized based on the data structure and non-linear relationships. Therefore, the splitting order in PLRTs may not reveal the same temporal sequence of an MMP. Thus, it is not feasible to select potential models for the prediction of the final product quality at an intermediate stage based only on the data available in the upstream stages, since the downstream variables may be needed to make the prediction. This limitation means that control or adjustment decisions cannot be made at an intermediate stage to reduce process variation in a MMP.
2. A PLRT model is usually used to predict a single response. Examples of multiple responses can be found in Segal (1992), Larsen and Speckman (2004), and Lee (2006) but not in a nested structure; i.e., where one response becomes a predictor to another response. In a variation reduction problem, an intermediate quality variable may be a response as well as a predictor to the downstream process. In a typical MMP, multiple variables need to be predicted for quality control purposes. However, it is difficult to evaluate the splitting conditions from multiple trees, which limits the ability to make a control or adjustment decision to achieve optimal performance of multivariate responses.

This article develops a unified modeling and control methodology for MMPs based on a reconfigured PLRT model. Engineering design knowledge is used to reconfigure the model into an engineering-compliant yet statistically equivalent model for feedforward control purposes. Furthermore, the model complexity is reduced by merging the splitting structures while satisfying the specified control accuracy requirement. Finally, a control strategy with an intermediate variable adjustment based on this reconfigured PLRT is proposed to reduce the variation in quality variables at the final stage.

The rest of this article is organized as follows. In Section 2, we propose the methodology for modeling the system and a feedforward control strategy. In Section 3, we use a multistage wafer manufacturing process to illustrate the procedure of modeling and control. Finally, conclusions are drawn in Section 4.

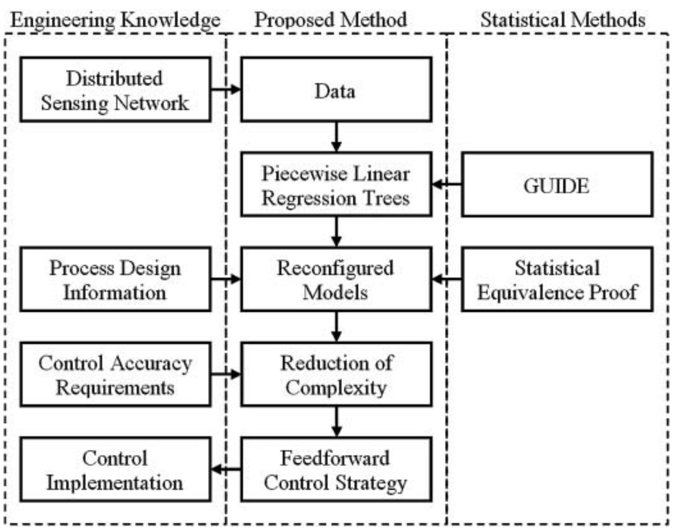

Fig. 1. Overview of the proposed methodology.

2. Reconfigured PLRT and feedforward control methodology

2.1. *Overview of the proposed methodology for modeling and control*

The proposed method to model and control an MMP with a reconfigured PLRT is an engineering knowledge–enhanced statistical method, as illustrated in Fig. 1.

In Fig. 1, the observational data of the process, material property, and quality variables are measured for an MMP. Based on these data, PLRTs are estimated using GUIDE that allow all intermediate and final quality variables to be predicted. The tree models are reconfigured to an engineering compliant structure with a statistically equivalent property. The reconfigured PLRT model structure is further adjusted based on the final quality specifications of the MMP to find the simplest model that satisfies the accuracy requirements. In the reconfigured PLRT, a group of potential prediction models is used to predict the final product quality as the multistage operations move from upstream stages to downstream stages. Therefore, a feedforward control strategy with intermediate process variable adjustment is used to take advantage of the temporally ordered layers in predicting quality variables. The control actions are iteratively determined by solving optimization problems using product and process constraints, which are conducted to improve the final product quality in the MMP.

2.2. *Engineering-driven reconfiguration of PLRTs*

The engineering-driven reconfiguration ensures the feasibility of the PLRTs in a feedforward control strategy. The advantage of the PLRTs in prediction accuracy is also preserved in control because the reconfiguration does not re-estimate the local models.

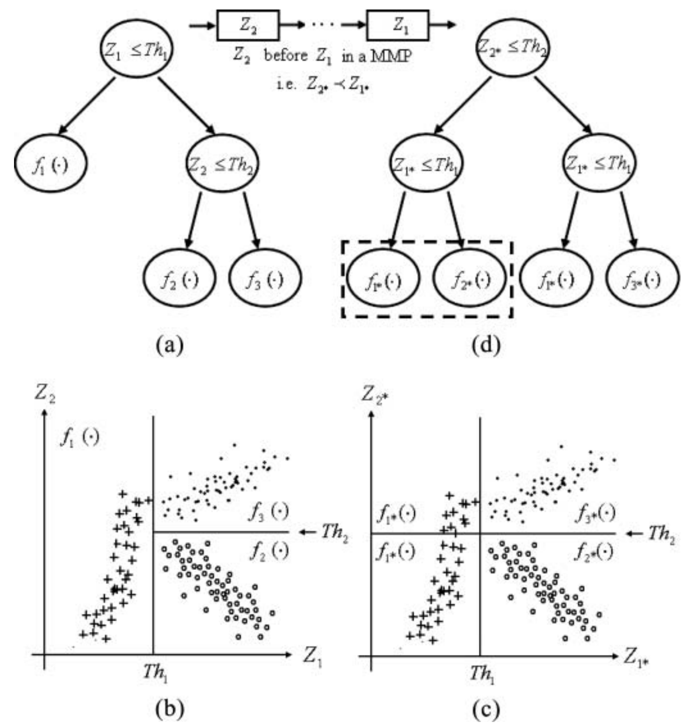

Fig. 2. Reordered model from a PLRT at one stage: (a) a PLRT, (b) sample space before reordering, (c) sample space after reordering, and (d) the reordered PLRT.

2.2.1. *MMP modeled by PLRTs*

PLRTs model the non-linear data by partitioning and local fitting. Figure 2(a) shows an example of a PLRT estimated from GUIDE, which consists of three leaf nodes. In this tree structure, $Z_i (i = 1, 2)$ are splitting variables, $Th_i (i = 1, 2)$ are splitting boundaries, and $f_i(\cdot) (i = 1, 2, 3)$ are local regression models. When the splitting condition holds, the tree goes to the left branch. The sample space of the PLRT is illustrated in Fig. 2(b), where $f_i(\cdot) (i = 1, 2, 3)$ are marked in their corresponding sub-regions.

In a typical MMP layout, as shown in Fig. 3, a stage is defined as a series of operations applied to a product to complete a manufacturing task. The intermediate quality variables are measured at each stage for modeling. A discrete part or a batch of products is processed. In this MMP, the variables can be classified as quality variables, process variables, or material property variables. The variables used in this approach are classified and summarized in Table 1.

To model the variable relationship, a PLRT is adopted by conducting regression of the quality variables on their upstream variables. A general form of the model with T leaf nodes and L distinct splitting variables can be written as

$$y = f(\boldsymbol{\eta}) = \sum_{i=1}^{T} f_i(\boldsymbol{\eta}_i) I(g_i(Z_1, \ldots, Z_L)). \qquad (1)$$

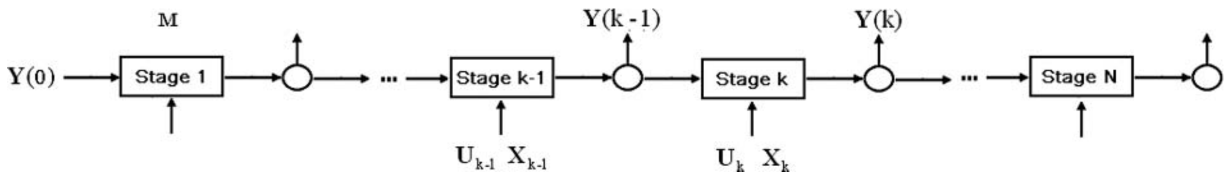

Fig. 3. A typical layout of an MMP.

In this model, y could be any quality variable at any stage; if y is a quality variable at the kth stage, then $\boldsymbol{\eta} = \{\mathbf{Y}(0), \mathbf{Y}(k_1), \mathbf{U}_{k_2}, \mathbf{X}_{k_2}, \mathbf{M}\}(k_1 = 1, 2, \ldots, k-1; \quad k_2 = 1, 2, \ldots, k)$ represents the known information at the kth stage; $f_i(\cdot)$ and $\boldsymbol{\eta}_i$ represent the local models and the covariates in the ith leaf node; $I(\cdot)$ is an indicator function, which has a value of one if $g_i(\cdot)$ is non-negative or zero otherwise; $g_i(\cdot)$ is the combination of conditions leading to the ith leaf node; and $Z_1, \ldots, Z_L$ are splitting variables for the tree structure. Furthermore, the $I(g_i(\cdot))$ can be decomposed into a product of the indicator functions of the individual splitting variables; i.e., $I(g_i(Z_1, \ldots, Z_L)) = \prod_{k=1}^{L} I(g_{i,k}(Z_k))$, where $g_{i,k}(\cdot)$ is the splitting condition of the kth variable for the ith leaf node. For example, in Fig. 2(a), the splitting conditions leading to $f_2(\cdot)$ are $Z_1 > Th_1$ and $Z_2 \leq Th_2$, which can be written as $I(g(Z_1, Z_2)) = I(Z_1 - Th_1)I(Th_2 - Z_2)$.

In the PLRT model estimation, there are three important issues to be addressed: (i) splitting variable selection; (ii) splitting boundary estimation; and (iii) tree structure determination. In this article, we follow the procedures in GUIDE, which recursively partitions the sample space, selects the splitting variables by contingence table test, and then determines the splitting boundaries by minimizing the prediction errors. When a large tree grows, the tenfold cross-validation error is minimized to prune the tree structure. There are comprehensive discussions on splitting variable selection, splitting boundary estimation, and pruning in the literature (Loh, 2002; Kim *et al.*, 2007; Loh *et al.*, 2007), which will not be repeated in this article. Those methods are used in this article to estimate a PLRT model from observational data. This estimated PLRT model will be used as a basis for later model reconfiguration and feedforward control design.

Table 1. Variable notations

$\mathbf{Y}(k) \in \Re^{m_k \times 1}$	: Quality variables with noise at the kth stage
$\mathbf{Y}(0)$	: Initial quality vector before entering the manufacturing process
$\mathbf{U}_k \in \Re^{r_k \times 1}$	: Continuous online controllable variables at the kth stage
u_{lk}	: The lth variable at the kth stage, which can be adjusted during the operations at the kth stage
$\mathbf{X}_k \in \Re^{n_k \times 1}$	: Offline setting variables at the kth stage
x_{lk}	: The lth variable at the kth stage, which can be adjusted between the $(k-1)$th stage and the kth stage
$\mathbf{M} \in \Re^{t \times 1}$	: Material property variables independent of stages

To explain the relationship of nodes in the tree structure, the *layer of nodes* in a tree is defined.

Definition 1. The ith layer of nodes. The ith layer of nodes in a tree is a set of nodes with depth i; i.e., the nodes that have $(i-1)$ splits from the root of the tree, including leaf nodes and splitting nodes.

Definition 1 is illustrated in Fig. 2(a). There are three layers because the deepest leaf node from the node is reached by two splittings from the root of the tree: the splitting node of $Z_1 \leq Th_1$ is the root node, which forms the first layer of the tree; leaf node $f_1(\cdot)$ and the splitting node of $Z_2 \leq Th_2$ form the second layer of the tree; leaf nodes $f_2(\cdot)$ and $f_3(\cdot)$ form the third layer of the tree.

2.2.2. *Reconfiguration of trees*

The engineering knowledge of MMPs used in the reconfiguration consists of the temporal order and the inherent relationships among the variables; i.e., the quality at the current stage is only influenced by the upstream stages rather than the downstream stages. When there is an insufficient Markov property of the quality variables, prediction by all upstream variables may also improve the prediction accuracy, compared to modeling by only regressing on the quality at the last stage.

Assuming that there are L splitting variables, these splitting variables belong to certain stages of the MMP with a temporal order. This article uses notations $\prec$, $\sim$, or $\prec\sim$ of variables marked by a * in the superscript to describe the temporal order. Table 2 summarizes the temporal relationship of these variables, and Z_{i^*} $(i = 1, 2)$ is used to denote Z_i in a temporal order. In MMPs, such a kind of temporal order of the quality and process variables at the $(k-1)$th stage and the kth stage can be presented as:

$$\mathbf{X}_{(k-1)^*} \prec\sim \mathbf{U}_{(k-1)^*} \prec \mathbf{Y}((k-1)^*) \prec \mathbf{X}_{k^*} \prec\sim \mathbf{U}_{k^*} \prec \mathbf{Y}(k^*).$$

Table 2. Notation for the temporal orders

$Z_{1^*} \prec Z_{2^*}$	: Z_1 is temporally prior to Z_2
$Z_{1^*} \sim Z_{2^*}$	: Z_1 and Z_2 have the same temporal order
$Z_{1^*} \prec\sim Z_{2^*}$	: Z_1 is temporally prior to or the same as Z_2

With the temporal order of the splitting variable, the original PLRT is reordered into a temporally compliant tree, which is defined below for further analysis.

Definition 2. Temporally compliant tree. A tree is temporally compliant if the splitting variables in the tree are temporally ordered, which is defined by the MMP layout; i.e., if $Z_{i^*} \prec\sim Z_{j^*}$, then Z_{i^*} is in a closer layer or the same layer as the root compared to the location of Z_{j^*}.

The reconfigured PLRT needs to have three properties to allow it to be used in feedforward control: (i) the reconfigured PLRT should be a temporally compliant tree; (ii) several PLRTs are estimated to predict the intermediate and final quality, which should be combined into a single decision structure; and (iii) the reconfigured PLRT should be statistically equivalent to the PLRT models with a high prediction accuracy.

The reconfiguration of the PLRTs consists of two steps: (i) each PLRT is reconfigured according to the temporal order of the splitting variables, called *reordering*; and (ii) a group of PLRTs is combined into a reconfigured PLRT, called *combining*.

2.2.2.1. *Reordering.* Assuming that the splitting order in a PLRT is not consistent with the temporal order as $Z_{1^*} \prec\sim Z_{2^*} \prec\sim \cdots \prec\sim Z_{L^*}$, the procedure to reorder a PLRT is proposed in Algorithm 1.

Algorithm 1.

Step 1. Convert the PLRT to a summation of $f_i(\cdot)$ and $g_i(\cdot)$ as Equation (1).

Step 2. Partition the region of $g_i(\cdot)$ with respect to all splitting variables into the decomposed sub-regions $g_i^j(\cdot)(j = 1, \ldots, D_i)$; that is,

$$y = \sum_{i=1}^{T} f_i(\boldsymbol{\eta}_i) I(g_i(Z_1, \ldots, Z_L)) = \sum_{i=1}^{T}\sum_{j=1}^{D_i} f_i(\boldsymbol{\eta}_i) I\big(g_i^j(Z_1, \ldots, Z_L)\big)$$

Step 3. Merge the sub-regions $g_i^j(\cdot)$ and $f_i(\boldsymbol{\eta}_i)$ for Z_i ($i = 1, \ldots, L$) from Z_{L^*} to Z_{1^*}, if Merge Condition I is satisfied. The final reordered model is

$$y^* = \sum_{i=1}^{T^*} f_{i^*}(\boldsymbol{\eta}_{i^*}) I(g_{i^*}(Z_{1^*}, \ldots, Z_{L^*})).$$

Step 4. Formulate the layers into a temporally compliant tree based on the reordered model.

In Algorithm 1, all splitting variables $Z_i(\forall i)$ are considered in partitioning the regions in Step 2; $g_i^j(\cdot)$ are the decomposed sub-regions of $g_i(\cdot)$, where D_i is the total number of sub-regions considering all possible splits of $Z_i(\forall i)$. In Step 3, if Merge Condition I (defined below) is satisfied, the sub-regions will be merged; otherwise, no further merging is needed.

The Merge Condition I for Z_i in any two decomposed sub-regions j_1 and j_2 in leaf nodes i_1 and i_2 is $g_{i_1,k}^{j_1}(Z_k) = g_{i_2,k}^{j_2}(Z_k)(\forall k \neq i)$ and $f_{i_1}(\boldsymbol{\eta}_{i_1})$ is the same model as $f_{i_2}(\boldsymbol{\eta}_{i_2})$. Here the splitting conditions of the decomposed regions are $I(g_{i_1,i}^{j_1}(Z_i))\prod_{\forall k\neq i} I(g_{i_1,i}^{j_1}(Z_k))$ and $I(g_{i_2,k}^{j_2}(Z_i))\prod_{\forall k\neq i} I(g_{i_2,k}^{j_2}(Z_k))$. $f_{i_1}(\boldsymbol{\eta}_{i_1})$ and $f_{i_2}(\boldsymbol{\eta}_{i_2})$ are the associated local regression models. After the merging process, the splitting condition for the newly merged leaf node is $\prod_{\forall k\neq i} I(g_{i_1,k}^{j_1}(Z_k))$ (or $\prod_{\forall k\neq i} I(g_{i_2,k}^{j_2}(Z_k))$).

To illustrate Merge Condition I, the tree in Fig. 2(a) is reordered as an example. Following the procedure of Algorithm 1, there will be four partitioned sub-regions as shown in Fig. 2(c) after Step 2. In Step 3, assuming $Z_{2^*} \prec Z_{1^*}$, the subregions should be merged first by eliminating Z_{1*}. Considering the merge in the dashed rectangle in Fig. 2(d), their splitting conditions are $I(Th_1 - Z_{1^*})I(Th_2 - Z_{2^*})$ and $I(Z_{1^*} - Th_1)I(Th_2 - Z_{2^*})$. In this example, $I(g_{1,2}^1(Z_{2^*})) = I(g_{2,2}^1(Z_{2^*})) = I(Th_2 - Z_{2^*})$, but $f_{1^*}(\cdot)$ and $f_{2^*}(\cdot)$ are not the same. Therefore, Merge Condition I is not satisfied and these two leaf nodes cannot be merged. Once the reordered model is obtained, we can formulate Z_{2^*} in the first layer and then Z_{1^*} in the second layer.

Statement 1. Statistical equivalence in reordering. *The original PLRT is statistically equivalent to the reordered temporally compliant tree in prediction; i.e.,* $y = y^*$.

The proof of Statement 1 is presented in the Appendix. To illustrate the equivalence, Fig. 2(b) and Fig. 2(c) are compared. For a new sample, the local prediction models $f_i(\cdot)\,(i = 1, 2, 3)$ in Fig. 2(b) and $f_{i*}(\cdot)\,(i = 1, 2, 3)$ in Fig. 2(c) are identical, since the reordering does not re-estimate the local regression models.

2.2.2.2. *Combining.* After reordering, multiple PLRTs are combined into a single reconfigured tree to predict multiple quality variables. If there are N_1 reordered PLRTs, with T_n^* leaf nodes and L_n^* splitting variables in the nth tree ($n = 1, 2, \ldots, N_1$), the general form of these reordered models is denoted as

$$y_n^* = \sum_{i=1}^{T_n^*} f_{i^*}^n(\boldsymbol{\eta}_{i^*}^n) I\big(g_{i^*}^n(Z_{1^*}^n, \ldots, Z_{L_n^*}^n)\big), \tag{2}$$

where the notation is the same as in Equation (1) except for n for the nth tree. Furthermore, $Z_{1^*}, \ldots, Z_{L^*}$ are the splitting variables in all of these trees, with temporal order $Z_{1^*} \prec\sim Z_{2^*} \prec\sim \cdots \prec\sim Z_{L^*}$. The procedure to combine the reordered models is proposed in Algorithm 2.

Algorithm 2.

Step 1. Obtain the reordered structure for the models in the form of Equation (2).

Step 2. Decompose the $g_i^n(\cdot)$ into $g_i^{n,j}(\cdot)$ using the same approach as in Step 2 in Algorithm 1 considering all splitting variables in different PLRTs; that is,

$$y_n^* = \sum_{i=1}^{T_n^*} f_{i^*}^n(\boldsymbol{\eta}_{i^*}^n) I(g_{i^*}^n(Z_{1^*}^n, \ldots, Z_{L_n^*}^n)) = \sum_{i=1}^{T_n^*} \sum_{j=1}^{D_{n,i}^*} f_{i^*}^n(\boldsymbol{\eta}_{i^*}^n) I(g_{i^*}^{n,j}(Z_{1^*}, \ldots, Z_{L^*})).$$

Step 3. Merge the decomposed sub-regions $g_i^{n,j}(\cdot)$ using a similar procedure to that in Step 3 of Algorithm 1 if Merge Condition II is satisfied. The final combined model is

$$y_n^* = \sum_{i=1}^{T^*} f_{i^*}^n(\boldsymbol{\eta}_{i^*}^n) I(g_{i^*}^{\text{comb}}(Z_{1^*}, \ldots, Z_{L^*}))$$
$$(n = 1, 2, \ldots, N_1).$$

Step 4. Formulate the layers into temporally compliant tree based on the combined model.

In Algorithm 2, all splitting variables in these reordered trees are considered in the decomposition in Step 2. $D_{n,i}^*$ is the total number of decomposed sub-regions considering all possible splits of $Z_{i^*}(\forall i)$ from the ith leaf node in the nth tree. In Step 3, if Merge Condition II is satisfied, the sub-regions will be merged, and a group of N_1 regression models for multiple responses is formed. Otherwise, no further merging is needed.

The Merge Condition II for Z_{i^*} in two decomposed sub-regions j_1 and j_2 in leaf nodes i_1 and i_2 is $I(g_{i_1^*,k^*}^{n,j_1}(Z_{k^*}^n)) = I(g_{i_2^*,k^*}^{n,j_2}(Z_{k^*}^n))$, $(\forall k^* \neq i^*)$, and $f_{i_1^*}^n(\boldsymbol{\eta}_{i_1^*}^n)$ is the same model as $f_{i_2^*}^n(\eta_{i_2^*}^n)$. Here the splitting conditions of these two decomposed sub-regions are $I(g_{i_1^*,i^*}^{n,j_1}(Z_{i^*}^n)) \prod_{\forall k^* \neq i^*} I(g_{i_1^*,k^*}^{n,j_1}(Z_{k^*}^n))$ and $I(g_{i_2^*,i^*}^{n,j_2}(Z_{i^*}^n)) \prod_{\forall k^* \neq i^*} I(g_{i_2^*,k^*}^{n,j_2}(Z_{k^*}^n))$. $f_{i_1^*}^n(\boldsymbol{\eta}_{i_1^*}^n)$ and $f_{i_2^*}^n(\boldsymbol{\eta}_{i_2^*}^n)$ are the associated local models. After the merging process, the splitting condition for the newly merged leaf node is $\prod_{\forall k^* \neq i^*} I(g_{i_1^*,k^*}^{n,j_1}(Z_{k*}^n))$ (or $\prod_{\forall k^* \neq i^*} I(g_{i_2^*,k^*}^{n,j_2}(Z_{k*}^n))$).

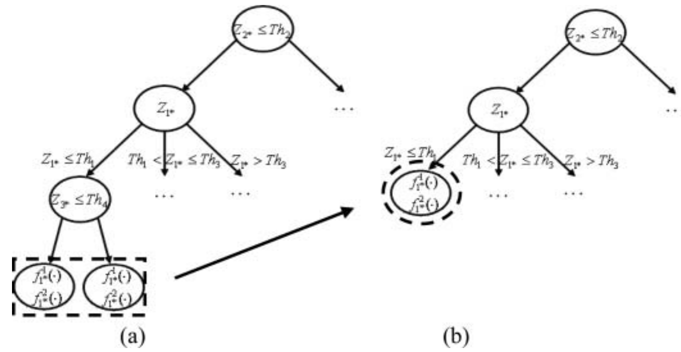

Fig. 5. Merging leaf nodes in combining: (a) before the merge and (b) after the merge.

To illustrate Merge Condition II, the two trees in Fig. 2(d) and Fig. 4 are combined as an example, assuming $Th_1 < Th_3$. In this case, the local models $f_{i^*}(\cdot)\,(i = 1, 2, 3)$ in Fig. 2(d) become $f_{i^*}^1(\cdot)\,(i = 1, 2, 3)$ to distinguish the models in Fig. 4. There are three distinct splitting variables in these trees: Z_{1^*}, Z_{2^*}, and Z_{3^*}. Following the procedure of Algorithm 2, all possible splits are generated in Step 2. In Step 3, assuming $Z_{2^*} \prec Z_{1^*} \prec Z_{3^*}$, the subregions should be merged first by eliminating Z_{3^*}. Considering the merger of two leaf nodes that are marked by the dashed rectangle in Fig. 5(a), the splitting conditions are $I(Th_1 - Z_{1^*})I(Th_2 - Z_{2^*})I(Th_4 - Z_{3^*})$ and $I(Th_1 - Z_{1^*})I(Th_2 - Z_{2^*})I(Z_{3^*} - Th_4)$. In this example, $I(g_{1,1}^{n,1}(Z_{1^*})) = I(g_{2,1}^{n,1}(Z_{1^*})) = I(Th_1 - Z_{1^*})$ for $n = 1, 2$, and $I(g_{1,2}^{1,1}(Z_{2^*})) = I(g_{2,2}^{1,1}(Z_{2^*})) = I(Th_2 - Z_{2^*})$. The local models are also identical. Therefore, the Merge Condition II is satisfied and these two leaf nodes should be merged, shown in Fig. 5(b).

Statement 2. Statistical equivalence in combining. A group of reordered models from PLRTs is combined into a single statistically equivalent model using Algorithm 2.

The proof of Statement 2 is shown in the Appendix. To illustrate the equivalence, the local models in the reordered trees (Fig. 2(d) and Fig. 4) are compared with the reconfig-

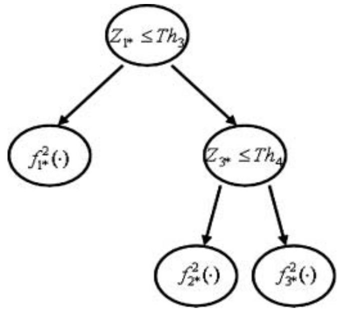

Fig. 4. Another reordered PLRT.

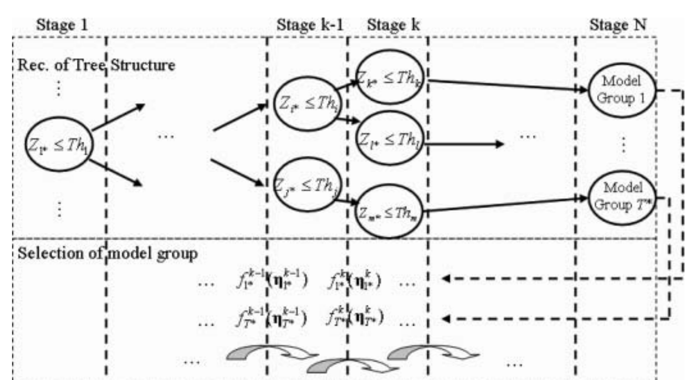

Fig. 6. Reconfigured PLRT for an MMP.

ured tree (Fig. 5(b)). For example, if $Z_{1^*} \leq Th_1$, $Z_{2^*} \leq Th_2$, and $Z_{3^*} > Th_4$, the local models for prediction are $f_{1^*}^1(\cdot)$ and $f_{1^*}^2(\cdot)$, which are the same as for the models with the same splitting conditions, shown by dashed lines in Fig. 5(b).

After the reconfiguration, the splitting variables are reordered into different layers, which map to the temporal order of the manufacturing stages, as shown in Fig. 6. The splitting conditions are combined, which leads to different model groups to predict the intermediate and final quality variables stage by stage. This reconfigured PLRT is preferred over the original PLRT for the feedforward control.

2.3. *Reconfigured model complexity and control accuracy*

The PLRTs from GUIDE are pruned by cross-validation to minimize the predicted Sum of Squared Errors (SSE; Loh, 2002). After the reconfiguration, the reconfigured model yields the best prediction accuracy due to the statistical equivalency. However, the reconfigured PLRT may be very complex, with many leaf nodes and many potential local models, which increases computational efforts in the control optimization. On the other hand, there is an engineering tolerance for the controlled objectives, which can be further transferred to the needs of the model precision used in the feedforward control. In other words, the model used for control may not have the same level of high precision requirement as the prediction obtained from the original PLRT. Therefore, the model complexity can be reduced, while the model still satisfies the control accuracy requirements. The reduction of the model complexity is achieved by assuming that there are a limited number of variables with a non-linear relationship with the response. A detailed discussion on how to further simplify the reconfigured PLRT with fewer leaf nodes is now provided.

In a reconfigured PLRT, the control performance can be evaluated by the accumulative errors of all PLRT model errors at different stages. However, different model groups may be selected for the control task based on the splitting conditions. Thus, it is difficult to estimate the control accuracy for every possible path used in the control task. In this article, the largest prediction variance is proposed to evaluate the control accuracy of this leaf node, which can be written as

$$\sigma_{k,j}^2 = \max_{\mathbf{U}_1,\ldots,\mathbf{U}_N,\mathbf{X}_1,\ldots,\mathbf{X}_N} \mathrm{Var}(\mathbf{Y}(N)_j)$$
$$\text{s.t. } x_{lk} \in \{x_{lk}\}, \quad u_{lk}^{\mathrm{L}} < u_{lk} < u_{lk}^{\mathrm{U}}, \forall l, \forall k \tag{3}$$

where $\sigma_{k,j}^2$ is the maximum prediction variance of the jth quality variable in the kth leaf node, obtained by enumerating all control actions; $\mathbf{Y}(N)_j$ is the predicted final quality variable; the optimization constraints s are the controllability of the process variables, where $\{x_{lk}\}$ is the set of all possible settings of x_{lk}; and u_{lk}^L, u_{lk}^U represents the lower and upper bound of the feasible range for u_{lk}.

The control accuracy of the overall structure is evaluated by the pooled variance of these leaf nodes. Assuming that there are equal numbers of products in different leaf nodes in the control task, the pooled variance is the average of the control accuracy of all leaf nodes; that is,

$$\sigma_{\mathrm{Rec.},j}^2 = \frac{1}{T}\sum_{k}^{T} \sigma_{k,j}^2, \tag{4}$$

where T is the number of leaf nodes in the reconfigured PLRT, and $\sigma_{\mathrm{Rec.},j}^2$ represents the control accuracy for the jth quality variable. Thus, the control accuracy can be evaluated by $\sigma_{\mathrm{Rec.},j}^2$.

The leaf nodes should be merged in order to reduce the model complexity. With fewer leaf nodes, the prediction performance will be degraded because the PLRTs are pruned to minimize the predicted SSE in the cross-validation. There are two issues to be addressed to balance the model complexity and the control accuracy: (i) which leaf nodes should be merged and (ii) when the merging process should be stopped.

The leaf nodes with the least important splitting structure should be merged first because this will result in the smallest decrease in the prediction accuracy. Although the control accuracy is evaluated based on the reconfigured PLRT, the temporally compliant splitting variables no longer provide information on the importance of the splitting structure. Nevertheless, the original PLRTs preserve the importance of the splitting variables for prediction in splitting orders from the most significant ones to the least significant ones. Therefore, reducing the number of leaf nodes will merge the nodes in the deepest layer in the original PLRTs. The merging process is stopped when the control accuracy of the reconfigured PLRT exceeds the predetermined control accuracy requirement.

The merging process is completed in an iterative way as shown in Fig. 7. In this figure the control accuracy of the current reconfigured PLRT $\sigma_{\mathrm{Rec.},j}^2$ is estimated first. Then the deepest leaf nodes in the original PLRTs are merged one at a time. In this way, a set of new control accuracy estimates $\sigma_{\mathrm{Rec.},j}^2$ of the final reconfigured PLRTs is

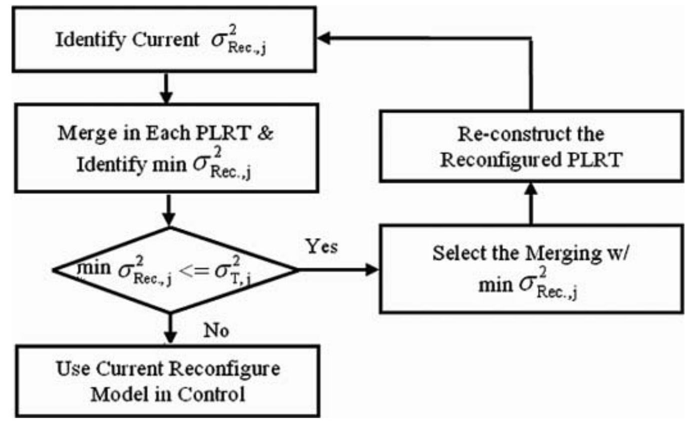

Fig. 7. The procedure to reduce model complexity.

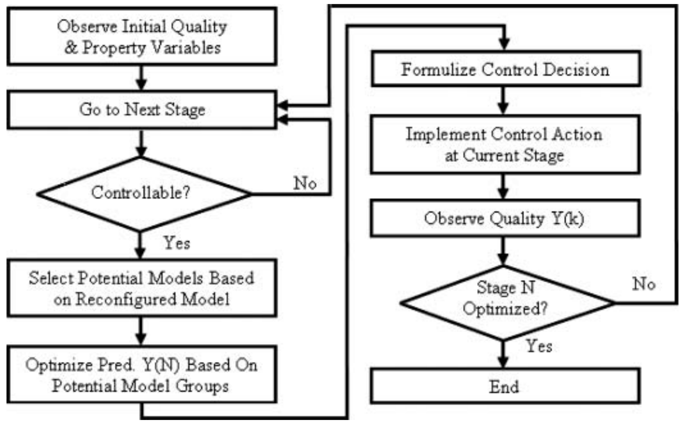

Fig. 8. The overall feedforward control strategy.

obtained. We choose the minimal $\sigma^2_{\text{Rec.},j}$ and compare it with a predetermined threshold $\sigma^2_{\text{T},j}$ for the jth quality variable. The model with the minimum $\sigma^2_{\text{Rec.},j}$ is acceptable if it is smaller than $\sigma^2_{\text{T},j}$. In this case, we reconstruct the reconfigured PLRT in the next iteration. Otherwise, the control accuracy of the current model does not satisfy the control accuracy requirement; thus, the merging operation should be stopped. After this procedure, the reconfigured model has reached a balance between the model complexity and the control accuracy.

2.4. *Feedforward control strategy of the reconfigured PLRT*

The engineering-driven reconfiguration makes it possible to develop a feedforward control strategy by actively adjusting the process variables and compensating the quality variable for variation reduction. The overall strategy is shown in Fig. 8. The basic idea is to achieve feedforward control based on the reconfigured PLRT models and how it is achieved is discussed in this section.

At each controllable stage, several potential model groups are determined based on the splitting conditions. If the splitting variables are measured at previous stages or layers, a model group in a leaf is selected when the splitting conditions are satisfied. Otherwise, several branches and leaves may be selected, which form a cluster of potential model groups. In this case, the splitting conditions are formulated as constraints in the optimization problem.

The control optimization at the kth stage is formulated as a smaller-the-better problem:

$$\begin{aligned}
&\min_{u_{li},x_{li},i=k,\dots N} J(\mathbf{U},\mathbf{X}) = \sum_{j=1}^{m} c_j\, E(\mathbf{Y}(N)_j^2),\\
&\text{s.t.}\quad \mathbf{Y}(N)_j = f_w^j(\boldsymbol{\eta}),\\
&\qquad h(\mathbf{Y}(s)_j) < H_{js},\\
&\qquad x_{li} \in \{x_{li}\},\\
&\qquad u_{li}^L < u_{li} < u_{li}^U,\\
&\qquad I(g_w(Z_1,\dots,Z_L)) > 0,\\
&\qquad s = 1,2,\dots,N;\ i = k,\dots,N,
\end{aligned} \tag{5}$$

where the objective function is the weighted sum of the second-order moment of m predicted final quality variables, $\mathbf{Y}(N)_j$ is the jth final quality variable predicted from the kth stage, and c_j is the weight of the importance of the jth quality variable. The decision variables are the process variables from the kth stage to the Nth stage. In the constraints, $f_\omega^j(\cdot)$ is a potential model group for the quality prediction determined by the splitting conditions; $h(\mathbf{Y}(s)_j) < H_{js}$ represents the quality specification for the jth quality variable at the sth stage ($s = 1, 2, \dots, N$); and $u_{li}^L < u_{li} < u_{li}^U$ and $x_{li} \in \{x_{li}\}$ ($i = k, \dots, N$) represent the controllability as described in Equation (3). The optimization problem is solved using the iterated local search algorithm (Stutzle, 1998).

3. Case study

A case study of a Multistage Wafer Manufacturing Process (MWMP) is conducted to illustrate the procedure of modeling and control based on the reconfigured PLRTs. A comparison study of the feedforward control strategy based on a reconfigured PLRT and regression model groups is conducted to show the effectiveness of the proposed approach.

3.1. *Wafer manufacturing processes*

An MWMP is a complex MMP involving chemical and mechanical processes that transform a silicon ingot into a wafer with uniform thickness, fine surface roughness, and good overall geometric shape for future processing. The process in this case study consists of five major manufacturing stages as shown in Fig. 9, including slicing, lapping, Chemical Vapor Deposition (CVD) of polysilicon, CVD of SiO_2, and polishing. Each stage is a combination of multiple operations, with quality measured at the end of the stage.

In an MWMP, the overall shape is a critical geometric quality index of a wafer. BOW and WARP of a wafer represent the overall shape of a wafer, which is used as the quality improvement objective in the case study. In general, smaller absolute values of these variables indicate a better quality for the wafer.

In this case study, observational data of three types of variables (quality, process, and material properties) were collected in a real production environment. Those variables are summarized in Table 3. In this table, the CTRRES represent the position of wafers in an ingot. In the case study, the central thickness of a wafer was measured at each stage, which was used in the selection of settings of downstream process parameters. Therefore, the central thickness of a wafer was treated as a predictor rather than a quality variable. The initial quality vector $\mathbf{Y}(0)$ in this process was assumed to be a zero vector.

In this process, some intermediate quality specifications of wafers need to be satisfied. For example, the thickness of a wafer in a certain lapping batch should be within a specified range; otherwise, the wafer will break during the lapping. These intermediate quality specifications were

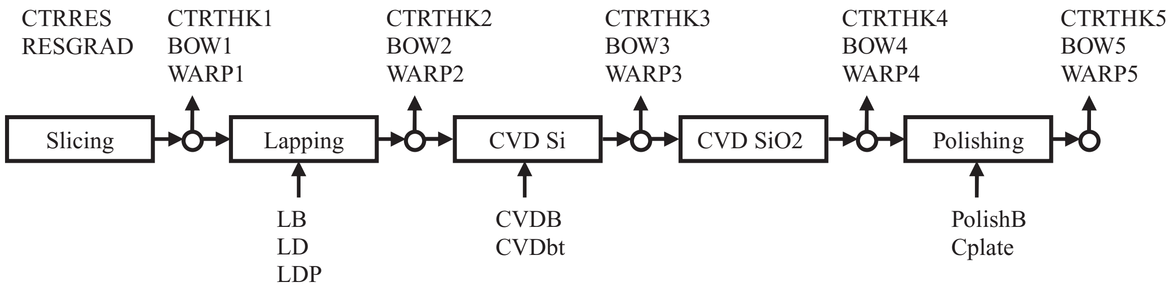

Fig. 9. A five-stage MWMP.

formulated as constraints in the optimization problem. Overall, data for 373 wafers were obtained in the production environment and were used in the case study. The PLRTs were constructed based on a training dataset (250 wafers) and the control performance was evaluated based on a testing dataset (123 wafers).

3.2. *PLRT models of the MWMP*

The PLRTs for this MWMP were estimated and are shown in Fig. 10. There are four splitting structures that have to be predicted, BOW2, BOW5, WARP2, and WARP5, while the models for other quality variables are regression models without splitting structures. In each leaf node, there is a local regression model, where "B" or "W" represents the quality variable BOW or WARP respectively. The PLRTs have explicit interpretations. For example, material property CTRRES at different segments of ingot yield different prediction models to predict BOW2 (Fig. 10(a)). This shows that the prediction of BOW2 is influenced by the material heterogeneity of wafers at the tail and the head of the ingot. Similar interpretations are obtained for BOW5, WARP2, and WARP5.

Since GUIDE did not consider the interactions in estimating the local regression models, the regression model of $f^{B3}(\cdot)$, $f^{B4}(\cdot)$, $f^{W3}(\cdot)$, and $f^{W4}(\cdot)$ was re-estimated considering the interactions of predictors to further reduce the predicted SSE.

Table 3. Measured variables in the MWMP

Variable type	*Variable name*	*Discrete/continuous*	*Measured stage*	*Physical meaning*
Process variables	LB	Discrete	Lapping	Lapping batch, representing processing time and compressive force with 15 levels
	LD	Discrete	Lapping	Lapping disk, representing maintenance conditions of pulley discs with five levels
	LDP	Discrete	Lapping	Position of wafer in lapping disks with six levels
	CVDB	Discrete	CVD Si	CVD batch, representing different tubes and processing time with five levels
	CVDbt	Discrete	CVD Si	CVD boat, representing wafers' positions in CVD tube
	PolishB	Discrete	Polishing	Polishing batch, representing age of slurry and polishing pad with 12 levels
	Cplate	Discrete	Polishing	Ceramic plate, representing the alignment of ceramic plate holders with four levels
Material property variables	CTRRES	Continuous	Independent of stages	Central resistivity of a wafer
	RESGRAD	Continuous	Independent of stages	Resistivity gradient of a wafer
Quality variables	BOW	Continuous	All five stages	Local warp at the center of a wafer
	WARP	Continuous	All five stages	Maximum local warp
	CTRTHK	Continuous	All five stages	Central thickness of wafer, used as predictors rather than responses

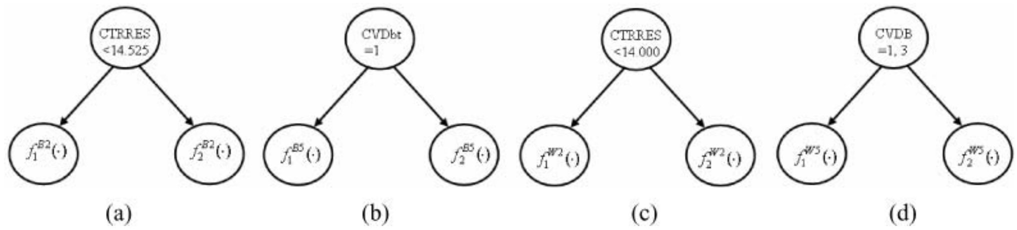

Fig. 10. PLRTs in MWMP: (a) model BOW2; (b) model BOW5; (c) model WARP2; and (d) model WARP5.

3.3. *Reconfiguration of PLRT*

Based on the PLRT from GUIDE, a reconfigured PLRT was obtained and is shown Fig. 11. In this figure, the temporal order of the splitting variables is CTRRES $\prec$ CVDB $\prec$ CVDbt. This order was split into different layers of the reconfigured PLRT from the root. In this example, CTRRES was split into three sub-regions in the first layer of the model, which were based on the splittings in the original models to predict BOW2 and WARP2. In the second and the third layers of the model, CVDB and CVDbt were split to be in the same manner as in the original model. In this way, 12 regression model groups were generated, which could be selected by the splitting conditions. The overall structure clearly represents the sequence of manufacturing from the root to the leaf nodes and predicts multiple intermediate and final quality variables.

3.4. *Reduce model complexity*

To reduce the model complexity, the control accuracy of BOW5 and WARP5 were evaluated in different reconfigured models. Figure 12(a) shows the control accuracy of 11 models from the regression group (Reg. G.) with the worst control accuracy to the reconfigured PLRT (Rec. T.) with the best control accuracy. The number of nodes is marked for each model. In this figure, the control accuracy varies as different model complexities are adopted. Such an analysis provides guidelines to select a model with an appropriate complexity that satisfies the control accuracy requirement. In this case study, the control accuracy requirement of BOW5 and WARP5 was taken to be 0.5 and 2.5 (horizontal dashed lines). The model with splits in BOW5 and WARP5 (B5W5) has the minimal number of leaf nodes to satisfy the requirement, which has only two significant splitting variables and four leaf nodes retained for control optimization, shown in Fig. 12(b). By comparing the "Rec. T." model, the model complexity has been significantly reduced.

3.5. *Simulation study of feedforward control*

To compare the feedforward control strategy, a total of 50 simulation runs were conducted based on three different models: Reg. G., B5W5, and Rec. T. In the simulation, the Reg. G. model is a global regression model without using splitting variables. The B5W5 and Rec. T. models use the reconfigured PLRT models for prediction. Without loss of generality, we set $c_j = 1$ in Equation (5).

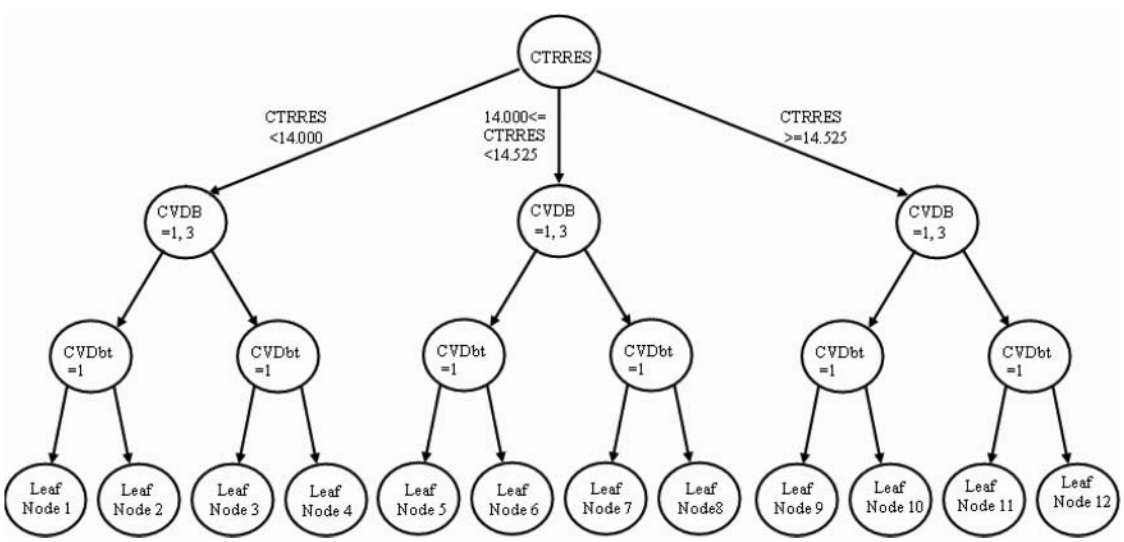

Fig. 11. Reconfigured PLRT for MWMP.

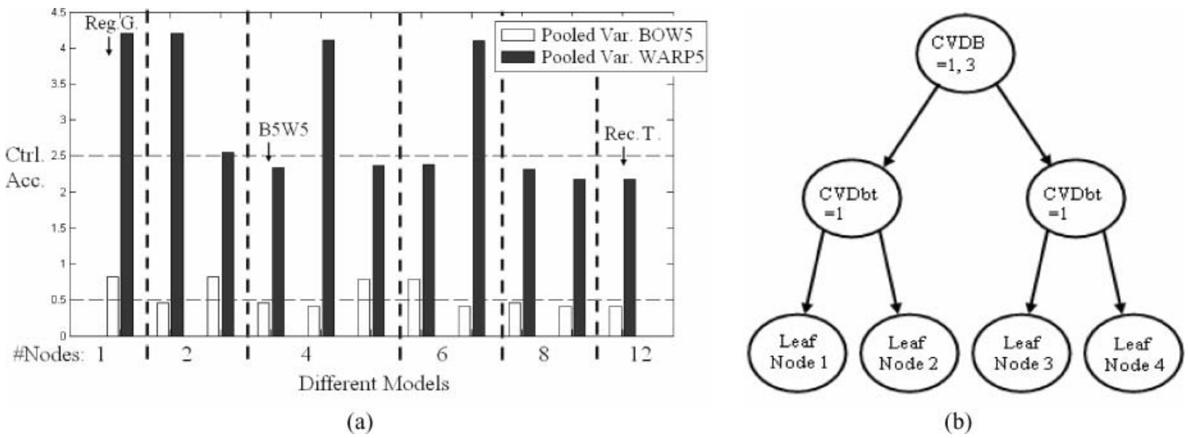

Fig. 12. Control accuracy and model complexity: (a) control accuracy of different models and (b) final reconfigured PLRT for control.

Figure 13(a) shows the controlled WARP5 in one simulation run. The horizontal axis represents the performance without control and with control based on different models. The control based on the reconfigured PLRTs yields better performance in reducing mean and variance of the final quality than regression group models. Moreover, there is no significant increase in mean and variance of the controlled quality when using the B5W5 model versus the Rec. T. model. This indicates that there is no significant loss in control performance when merging some of the splitting structures. Figure 13(b) shows the controlled performance of the absolute value of BOW5 with similar interpretations.

The values of the optimal objective function of 50 simulation runs are summarized in Fig. 14. The values of the optimal objective function based on Reg. G. are larger than those based on reconfigured PLRT in most of the simulation runs the; i.e., better control performance is obtained with the reconfigured PLRT model. The Rec. T. model has a better controlled performance than the B5W5 model. However, a more complex model structure leads to a higher demand on computational efforts. The proposed reconfigured PLRT with reduced model complexity has fewer leaf nodes and thus sacrifices control accuracy, but it still sufficiently meets the control requirements from an engineering perspective.

Table 4. Controlled objective values in simulations

	Reg. G.	*B5W5*	*Rec. T.*
Mean	383.80	340.68	307.05
St. Dev.	35.63	7.64	15.20

The mean and standard deviation of the optimal values are summarized in Table 4. There is an average of 11.24 and 20.00% reduction in objective value for the B5W5 and Rec. T. compared to Reg. G. The standard deviation of the values of the objective function is also reduced for the proposed B5W5 model. The study indicates that the reconfigured PLRT is more effective in variation reduction than the standard regression models based on the proposed control strategy.

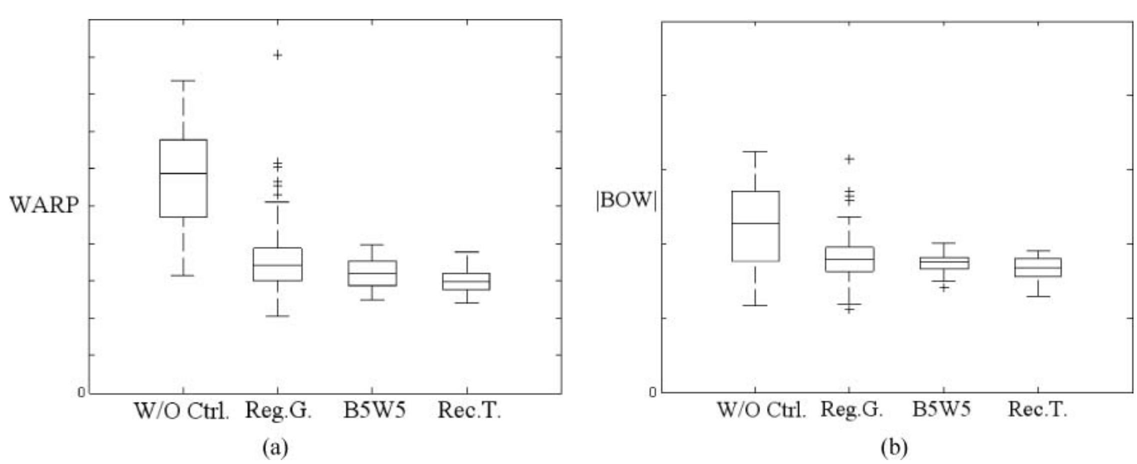

Fig. 13. Controlled quality performance in a simulation run: (a) WARP5 and (b) |BOW5|.

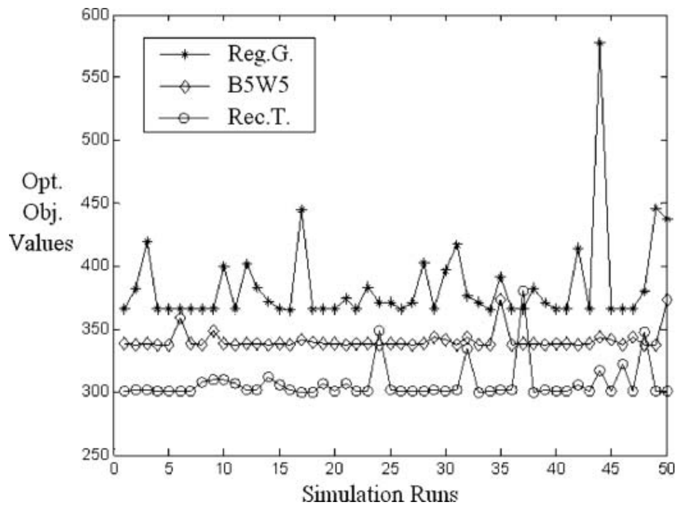

Fig. 14. Comparison of control performance based on different models.

4. Conclusions

It is a challenging task to model the variations and their propagation in MMPs, especially when the relationships among process parameters and product quality variables are non-linear. In this case, a PLRT model can be adopted that has high prediction accuracy and explicit interpretation in describing non-linear data structures. However, it fails to illustrate the temporal order and inherent relationships among variables in a MMP.

This article bridges the gap between the needs for advanced models for MMP variation reduction and the limitations of PLRT. An engineering-driven reconfiguration of the PLRT is proposed to convert the original model into an engineering compliant model. The reconfigured PLRT not only has a high prediction accuracy of the original tree structure but also provides a feasible solution in determining the potential prediction models sequentially as the operations move from the upstream stages to the downstream stages. This sequential model selection procedure enables its capability in active compensation by implementing a feedforward control strategy. The model complexity is also reduced by analyzing the control accuracy of the models. A case study has been conducted in a real MWMP, which demonstrates better control performance by using the reconfigured PLRT model compared to that using a standard regression model.

Acknowledgement

The authors gratefully acknowledge the support of National Science Foundation under grant NSF CMMI-0927574.

References

Breiman, L., Friedman, J.H., Olshen, R.A. and Stone, C.J. (1984) *Classification and Regression Trees*, Wadsworth, Belmont, CA.

Chaudhuri, P., Huang, M.C., Loh, W.Y. and Yao, R. (1994) Piecewise-polynomial regression trees. *Statistical Sinica*, **4**, 143–167.

Chipman, H., George, E. and McCulloch, R. (1998) Bayesian CART model search (with discussion). *Journal of the American Statistical Association*, **93**, 935–960.

Chipman, H., George, E. and McCulloch, R. (2002) Bayesian tree models. *Machine Learning*, **48**, 299–320.

Denison, D., Adams, N., Holmes, C. and Hand, D. (2002) Bayesian partition modeling. *Computational Statistics and Data Analysis*, **38**, 475–485.

Djurdjanovic, D. and Zhu, J. (2005) Stream of variation based error compensation strategy in multistation manufacturing processes, in *Proceedings of the 2005 ASME International Mechanical Engineering Congress and Exposition*, paper IMECE2005-81550, 314–319.

Izquierdo, L.E., Shi, J., Hu, S.J. and Wampler, C.W. (2007) Feedforward control of multistage assembly processes using programmable tooling. *Transactions of NAMRI/SME*, **35**, 295–302.

Jiao, Y. and Djurdjanovic, D. (2010) Joint allocation of measurement points and controllable tooling machines in multistage manufacturing processes. *IIE Transactions*, **42**, 703–720.

Jin, J. and Ding, Y. (2004) Online automatic process control using observable noise factors for discrete part manufacturing. *IIE Transactions*, **36**, 899–911.

Jin, J. and Shi, J. (1999) State space modeling of sheet metal assembly for dimensional control. *ASME Transactions*, *Journal of Manufacturing Science and Engineering*, **121**, 756–762.

Joseph, R. (2003) Robust parameter design with feed-forward control. *Technometrics*, **45**, 284–292.

Kim, H., Loh, W.Y., Shih, Y. and Chaudhuri, P. (2007) Visualizable and interpretable regression models with good prediction power. *IIE Transactions*, **39**, 565–579.

Larsen, D.R., and Speckman, P.L. (2004) Multivariate regression trees for analysis of abundance data. *Biometrics*, **60**, 543–549.

Lee, S.K. (2006) On classification and regression trees for multiple responses and its application. *Journal of Classification*, **23**(1), 123–141.

Loh, W.Y. (2002) Regression trees with unbiased variable selection and interaction detection. *Statistical Sinica*, **12**, 361–386.

Loh, W.Y. (2007) Regression by parts: fitting visually interpretable models with GUIDE, in *Handbook of Data Visualization*, Chen, C., Hardle, W. and Unwin, A. (eds), Springer-Verlag, Berlin, Germany, pp. 447–468.

Loh, W.Y., Chen, C. and Zheng, W. (2007) Extrapolation errors in linear model trees. *ACM Transactions on Knowledge Discovery in Data*, **1**, 1–17.

Morgan, J.N. and Sonquist, J.A. (1963) Problems in the analysis of survey data, and a proposal. *Journal of the American Statistical Association*, **58**, 415–434.

Segal, M.R. (1992) Tree-structured methods for longitudinal data. *Journal of the American Statistical Association*, **87**, 407–418.

Shi, J. (2006) *Stream of Variation Modeling and Analysis for Multistage Manufacturing Processes*, CRC Press, New York, NY.

Stutzle, T. (1998) Local search algorithms for combinatorial problems-analysis, improvements and new applications. Ph.D. thesis, FB Informatik, TU Darmstadt.

Zhong, J., Shi, J. and Wu, C.F. (2010) Design of DOE-based automatic process controller with consideration of model and observation. *IEEE Transactions on Automation Science and Engineering*, **7**, 266–273.

Appendix

Proof of Statement 1. The temporal order of the splitting variables is assumed as $Z_{1^*} \prec= Z_{2^*} \prec= \cdots \prec= Z_{L^*}$. In the decomposition of the sub-regions of $g_i(\cdot)$ into $g_i^j(\cdot)$:

$$y = \sum_{i=1}^{T} f_i(\eta_i) I(g_i(Z_1, \ldots, Z_L)) = \sum_{i=1}^{T}\sum_{j=1}^{D_i} f_i(\eta_i) I\big(g_i^j(Z_1, \ldots, Z_L)\big). \quad \text{(A1)}$$

Since the decomposed sub-regions involve all splitting variables, the temporally compliant variables can be substituted into $g_i^j(\cdot)$:

$$y = \sum_{i=1}^{T}\sum_{j=1}^{D_i} f_i(\eta_i) I(g_i^j(Z_{1^*}, \ldots, Z_{L^*})). \quad \text{(A2)}$$

Since the splitting variables are temporally compliant, the tree can be rearranged into a temporally compliant tree. Based on this tree, the merge of sub-regions follows the reverse temporal order. After the merge, sub-region $g_{i^*}^j(\cdot)$ is the jth region defined by a subset of $\{Z_{i^*}\}$ for $f_i(\cdot)$, and there are T^* leaf nodes left:

$$y = \sum_{i^*=1}^{T^*} f_{i^*}(\eta_{i^*}) I(g_{i^*}(Z_{1^*}, \ldots, Z_{L^*})) = y^*. \quad \text{(A3)}$$

The original PLRT is statistically equivalent to the reordered model in prediction.

This completes the proof. ■

Proof of Statement 2. Without loss of generality, consider the case when there are two reordered models to be combined together, which are $y_1^* = \sum_{i=1}^{T_1^*} f_{i^*}^1(\eta_{i^*}^1) I(g_{i^*}^1(Z_{1^*}^1, \ldots, Z_{L_1^*}^1))$ and $y_2^* = \sum_{i=1}^{T_2^*} f_{i^*}^2(\eta_{i^*}^2) I(g_{i^*}^2(Z_{1^*}^2, \ldots, Z_{L_2^*}^2))$. If $g_i^n(\cdot)$ is decomposed by all possible splits of the splitting variables in both models, then the first model is

$$y_1^* = \sum_{i=1}^{T_1^*} f_{i^*}^1(\eta_{i^*}^1) I\big(g_{i^*}^1(Z_{1^*}^1, \ldots, Z_{L_1^*}^1)\big)$$
$$= \sum_{i=1}^{T_1^*}\sum_{j=1}^{D_{1,i}^*} f_{i^*}^1(\eta_{i^*}^1) I\big(g_{i^*}^{1,j}(Z_{1^*}^1, \ldots, Z_{L_1^*}^1, Z_{1^*}^2, \ldots, Z_{L_2^*}^2)\big)$$
$$= \sum_{i=1}^{T_1^*}\sum_{j=1}^{D_{1,i}^*} f_{i^*}^1(\eta_{i^*}^1) I\big(g_{i^*}^{1,j}(Z_1, \ldots, Z_{L^*})\big). \quad \text{(A4)}$$

Similarly, the second model is

$$y_2^* = \sum_{i=1}^{T_2^*}\sum_{j=1}^{D_{2,i}^*} f_{i^*}^2(\eta_{i^*}^2) I\big(g_{i^*}^{2,j}(Z_1, \ldots, Z_{L^*})\big). \quad \text{(A5)}$$

Since all possible splits of the splitting variables in both models are considered:

$$g_{i^*}^{1,j}(Z_1, \ldots, Z_{L^*}) = g_{i^*}^{2,j}(Z_1, \ldots, Z_{L^*}). \quad \text{(A6)}$$

By following the procedure in Step 3 of Algorithm 2, these two models can be presented as.

$$y_1^* = \sum_{i^*}^{T^*} f_{i^*}^1(\eta_{i^*}^1) I\big(g_{i^*}^{\text{comb}}(Z_1, \ldots, Z_{L^*})\big) \quad \text{(A7)}$$

and

$$y_2^* = \sum_{i*}^{T^*} f_{i^*}^2(\eta_{i^*}^2) I\big(g_{i^*}^{\text{comb}}(Z_1, \ldots, Z_{L^*})\big), \quad \text{(A8)}$$

where $g_{i^*}^{\text{comb}}(Z_1, \ldots, Z_{L^*})$ in both models are the same. Therefore, the combined model is the same as the original two reordered models in the prediction. ■

Biographies

Ran Jin is a Ph.D. student of the H. Milton Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology. He received a B.Eng. in Electronics Information Engineering at the Tsinghua University, Beijing, in 2005, an M.S. in Industrial Engineering, and an M.A. in Statistics at the University of Michigan, Ann Arbor, in 2007 and 2009, respectively. His research interests include data mining, engineering knowledge–enhanced statistical modeling of complex systems, process monitoring, diagnosis, and control.

Jianjun Shi is the Carolyn J. Stewart Chair Professor at the H. Milton Stewart School of Industrial and Systems Engineering, Georgia Institute of Technology. Before joining Georgia Tech in 2008, he was the G. Lawton and Louise G. Johnson Professor of Engineering at the University of Michigan. He was awarded B.S. and M.S. degrees in Electrical Engineering at the Beijing Institute of Technology in 1984 and 1987, respectively, and a Ph.D. in Mechanical Engineering at the University of Michigan in 1992. His research interests focus on the fusion of advanced statistics, signal processing, control theory, and domain knowledge to develop methodologies for modeling, monitoring, diagnosis, and control for complex systems in data-rich environments. He is the founding chairperson of the Quality, Statistics and Reliability subdivision at INFORMS. He currently serves as the Focus Issue Editor of *IIE Transactions on Quality and Reliability Engineering*. He is a Fellow of the Institute of Industrial Engineering, a Fellow of the American Society of Mechanical Engineering, a Fellow of the Institute of Operations Research and Management Science, and a life member of ASA.