Sequential adaptive design for jump regression estimation

Chiwoo Park\textsuperscript{a}, Peihua Qiu\textsuperscript{b}, Jennifer Carpena-Núñez\textsuperscript{c}, Rahul Rao\textsuperscript{d}, Michael Susner\textsuperscript{e}, and Benji Maruyama\textsuperscript{c}

\textsuperscript{a}Department of Industrial and Manufacturing Engineering, Florida State University, Tallahassee, FL, USA; \textsuperscript{b}Department of Biostatistics, University of Florida, Gainesville, FL, USA; \textsuperscript{c}Air Force Research Laboratory, Wright-Patterson AFB, OH, USA

ABSTRACT

Selecting input variables or design points for statistical models has been of great interest in adaptive design and active learning. Motivated by two scientific examples, this article presents a strategy of selecting the design points for a regression model when the underlying regression function is discontinuous. The first example we undertook was to accelerate imaging speed in high-resolution material imaging, and the second was to use sequential design for mapping a chemical phase diagram. In both examples, the underlying regression functions have discontinuities, and thus many existing design optimization approaches cannot be used, as they assume a continuous regression function. Although some existing adaptive design strategies developed from the treed regression models can handle the discontinuities, the related Bayesian model estimation approaches come with computationally expensive Markov Chain Monte Carlo algorithms for posterior inferences and the subsequent design point selections, which may not be applicable for the first motivating example that requires the computation to be faster than the original imaging speed. In addition, the treed models are based on domain partitioning and are inefficient in cases when the discontinuities occur at complex sub-domain boundaries. In this article, we propose a simple and effective adaptive design strategy for regression analysis with discontinuities. After some statistical properties of the estimated regression model are derived in cases with a fixed design, a new criterion for sequentially selecting the design points is suggested. The suggested sequential design selection procedure is then evaluated using a comprehensive simulation study and demonstrated using two motivating examples.

1. Introduction

Regression analysis is a powerful statistical tool that can be used to estimate a regression function that relates explanatory variables to a response variable. In a typical regression analysis, the design points are assumed to be given in advance. When the design points can be selected during a data collection process, optimizing the selection is referred to as optimal design (Chernoff, 1972), active learning (Cohn et al., 1996), or adaptive sensing (Arias-Castro et al., 2013; Malloy and Nowak, 2014). This article aims to address the problem of selecting the design points for a regression model when the underlying regression function is piecewise continuous, which is motivated by two scientific applications described below.

The first motivating application is material imaging with Scanning Transmission Electron Microscopy (STEM). The STEM technique is a vital material characterization tool to image the microstructure of a material specimen at a fine spatial resolution. It uses a focused beam of electrons to probe a material specimen, and the intensity of the beam interacting with the specimen is measured for every focus location. This sequential imaging process creates a raster image of the material specimen as shown in Figure 1(a), where a pixel corresponds to one focus location over the raster, and the corresponding intensity is the pixel intensity. The radius of the focused beam can be below $10^{-10}$ m, which allows a specimen to be imaged at an excellent spatial resolution; however, this level of detail is also the primary reason for a slow imaging speed. Accelerating the STEM imaging speed would open unprecedented opportunities in studying important material processes (Park and Ding, 2019, 2021). The existing approaches to accelerate the imaging speed are based on a partial scan (i.e., scan a material specimen only at selected pixel locations). In the existing approaches (Stevens et al., 2015), the partial set is randomly selected from a uniform distribution over the raster locations, which can cause a loss in spatial resolution. Optimizing the pixel locations in the partial set is highly desirable to mitigate the information loss. In this example, the image intensity surface can be regarded as a 2D regression function of the pixel locations. This function would be piecewise continuous, as the image intensity can suddenly change at the boundary between different base materials of the imaging specimen, as shown in Figure 1(a). Selecting the pixel locations for the partial scan can be formulated as a design optimization problem for estimating the underlying 2D piecewise continuous regression function.

CONTACT Chiwoo Park \textsuperscript{a} cpark5@fsu.edu Copyright © 2021 “IISE”
The second motivating application is to optimize the design of experiments for effectively exploring a chemical phase diagram in chemistry. A phase diagram is a map that relates different experimental conditions to the physical states of materials. The physical state suddenly changes from one state to another around the experimental conditions where phase transitions occur, as illustrated in Figure 1(b). Typically, the elucidation of a phase diagram requires a large number of experiments to be performed to probe possible physical states that may exist in the experimental phase-space. Emerging trends in materials research are to use machine learning algorithms to achieve the data-driven optimization of experimental designs, namely, autonomous experimentation (Stach et al., 2021). In particular, we are interested in experiments to study the chemical conditions required for good carbon nanotube growth. The chemical conditions include the reaction temperature and a relative ratio of two chemical ingredients (i.e., a reducing agent and an oxidant). The total nanotube growth changes abruptly around the boundary condition in the relative ratio for a given temperature. Therefore, the total nanotube growth is a piecewise continuous function of the relative ratio and temperature. Optimizing the experimental design so as to relate the two conditions to the nanotube growth can be formulated as the problem of selecting the design points for estimating the piecewise continuous function.

In both motivating applications described above, the underlying regression functions are piecewise continuous in low-dimensional design spaces. In many existing studies of sequential design and active learning, however, the underlying regression function is often assumed to be continuous. Many design selection approaches have been developed under that assumption, as we will review in Section 2. This article develops a simple and efficient sequential adaptive design strategy for estimating a piecewise continuous regression function and presents numerical evaluations of this new strategy for regression problems with two or three input dimensions.

The remainder of this article is organized as follows. In Section 2, we review the existing research in active learning and sequential design and emphasize the need for a new adaptive design for our two motivating applications. Section 3 describes an approach of jump regression analysis for estimating a discontinuous regression function and discusses its statistical properties in cases with a fixed design. This approach is then used to develop a novel sequential adaptive design strategy for jump regression analysis. Section 4 presents numerical studies to evaluate the performance of the proposed method. Sections 5 and 6 illustrate the proposed method using the two motivating applications. Finally, Section 7 concludes the article with some summary statements.

2. Related work

The design optimization problem has been studied in the sequential adaptive design and active learning literature. Sequential adaptive designs and active learning are branched out from different disciplines, so they each have a slightly different focus. Sequential adaptive designs are studied in the design of experiments, where the designs are often optimized to improve the parameter estimates of statistical models (Sacks et al., 1989). Active learning originates from the machine learning community to optimize data collection to improve the accuracy of machine learning models (Cohn et al., 1996). Nonetheless, they are similar in that they both aim to optimize data acquisition to improve estimated statistical models.

Many experimental designs and active learning strategies have been developed for regression modeling with continuous regression functions. Sequential experimental design allows experiments to be conducted sequentially to exploit past experimental outcomes to guide the design of future experiments. Many sequential design strategies have been developed for parametric regression models (Chaudhuri and Mykland, 1993; Dror and Steinberg, 2008) and nonlinear regression models, using smoothing approaches such as
kernel regression (Zhao and Yao, 2012), wavelet regression (Bull, 2013), and Gaussian process regression (Zhu and Stein, 2006; Zimmerman, 2006). In active learning, selecting design points has been studied for many nonparametric regression models, such as Gaussian process regression models (Krause et al., 2008; Singh et al., 2009; Hoang et al., 2014) and kernel-based regression models (Paisley et al., 2010). These existing approaches are not designed to estimate regression models with discontinuous regression functions.

There are a few existing adaptive design strategies that can be used to estimate piecewise continuous regression functions. In these methods, the input domain is first partitioned into sub-domains, and then a simple regression model is fitted in each sub-domain. Depending on the ways of domain partitioning, we can group the existing methods into two groups: Bayesian tree-based approaches and Voronoi tessellation-based approaches. The Bayesian tree-based approaches recursively partition the input domain along one of the axis directions. Malloy and Nowak (2014) and Goetz et al. (2018) studied adaptive learning strategies using such approaches to estimate piecewise constant functions. Bull (2013) discussed an active learning strategy to estimate spatially inhomogeneous regression functions, including piecewise constant functions and functions with sharp bumps. However, this method was limited to cases with a single explanatory variable. Gramacy and Lee (2009) discussed a treed Gaussian Process (GP) model and the corresponding sequential design strategies using tree-partitioning and GP leaves. Its posterior inference involved computationally inefficient reversible-jump methods for Markov Chain Monte Carlo (MCMC) or higher-dimensional particles for sequential inference. Taddy et al. (2011) proposed a dynamic tree model with constant or linear leaves and developed an adaptive design strategy for the model. It does not involve MCMC sampling steps and thus is computationally more efficient.

There are also methods that use Voronoi tessellation-based domain partitioning to estimate regression models with piecewise continuous regression functions. The method in Kim et al. (2005) first partitions the input domain by the Voronoi tessellation of training inputs, and then models the regression surface in each sub-domain as a local GP. Pope et al. (2021) discusses an adaptive sampling strategy to learn the piecewise regression model. The major issue with the tessellation-based methods is that the Voronoi tessellation only works effectively in spatial domains and is computationally expensive in 3D. The tessellation in higher dimensions is known to be an NP-hard problem. It is not surprising that these works are only applied to 2D spatial modeling problems. There are active learning strategies for partitioned GP models (Cortes et al., 2019). They are focused on how to perform active learning when the domain partitioning is known or it can be easily estimated by a simple clustering approach. The estimation of the domain partition is a major complexity largely unaddressed by such approaches.

Another potentially related approach is the adaptive design to estimate the contour of the underlying regression function \( m(x) \) (Ranjan et al., 2008), which sequentially selects the design points used to estimate the contour of the subregion \( m(x) < c \), for some constant \( c \). When the discontinuities in \( m(x) \) occur at a constant level \( c \), estimating the contour at \( c \) would create a partition of the regression domain around the discontinuities, and thus allow the use of several regression models in sub-domains. The major limitation to using this approach in the two motivating applications is that the level \( c \) should be known a priori, and its estimation is not straightforward, as the discontinuities do not always occur at a single level \( c \) (e.g., material images on uneven backgrounds such as MG5 and MG6 in Figure 9 in the first motivating application).

To sum up, the piecewise regression modeling approaches explicitly partition the input domain to model separate regression functions in sub-domains. The joint learning of the partitioning and subregional model fitting often involves computationally expensive MCMC samplings. Dynamic trees (Taddy et al., 2011) are simple and more computationally feasible, but they are limited to piecewise constant or linear models. A more substantial issue around a piecewise regression model is that the explicit partitioning scheme is not flexible enough to efficiently model curvy discontinuities in the regression function. The tree-based models partition the input domain along one of the axis-aligned directions. The tessellation-based partitioning is applicable only for a spatial domain. As a comparison, jump regression analysis (Qiu, 2005) provides a simpler modeling approach for a broader class of piecewise continuous regression functions. The major novelty in this article is that we develop a simple and flexible nonparametric approach to the estimation of piecewise continuous regression functions, and a sequential adaptive design strategy for jump regression analysis. The new approach is applicable in higher dimensions and is more computationally efficient compared with the existing methods. The proposed modeling approach meets the time constraint and model adequacy issue of the two motivating applications.

3. Method

Let \( \mathcal{X} \) denote a closed subset of \( \mathbb{R}^p \) that represents a design space in a regression modeling problem. We consider a general jump regression model that aims to estimate a nonparametric regression function \( m: \mathcal{X} \rightarrow \mathbb{R} \) from its noisy observations that follow the model:

\[
Y_i := m(x_i) + \epsilon_i,
\]

where \( \{Y_i; i = 1, ..., n\} \) are noisy observations of the response variable \( Y \) at the design points \( \{x_i \in \mathcal{X}; i = 1, ..., n\} \), and \( \{\epsilon_i; i = 1, ..., n\} \) are random errors with mean zero and variance \( \sigma^2 \). The underlying regression function \( m \) is further assumed to be piecewise continuous such that
there exists a partition \(\{A_b; b = 1, ..., B\}\) of the design space \(\mathcal{X}\) satisfying:

1. Each \(A_b\) is a simple connected (nonempty) subset of \(\mathcal{X}\).
2. \(\bigcup_{b=1}^{B} A_b = \mathcal{X}\), and \(A_b \cap A_{b'} = \emptyset\), for any \(b \neq b'\).
3. The function \(m(x)\) has the expression:
   \[
   m(x) = \sum_{b=1}^{B} g_{b}(x) I_{A_b}(x), \quad \text{for } x \in \mathcal{X},
   \]
   where \(g_{b}(x) \in C^2(\mathcal{X})\) is a smooth function, for each \(b\).

   Thus, the regression function \(m(x)\) is continuous in \(A_b \setminus \partial A_b\), where \(\partial A_b\) is the boundary set of \(A_b\), and it has jumps over \(B := \bigcup_{b=1}^{B} \partial A_b\). For any \(x^* \in B\), there exists \(b\) and \(b'\) such that \(x^* \in \partial A_b \cap \partial A_{b'}\) and
   \[
   \lim_{x \to x^*, x \in A_b} g_{b}(x) \neq \lim_{x \to x^*, x \in A_{b'}} g_{b'}(x).
   \]

   The boundary set \(B\) consists of the Jump Location Curves (JLCs) of \(m\) in the literature (Qiu, 1998).

4. The boundary is smooth so that a tangent line exists almost everywhere. A point in the boundary set \(B\) is called non-singular when there exists a unique tangent line at the point. Otherwise, it is called singular. We denote a collection of all singular boundary points by \(S\). Then, for any \(x^* \in B \setminus S\), there exists a unique pair \((b, b')\) such that \(x^* \in \partial A_b \cap \partial A_{b'}\). Otherwise, its tangent line would not be unique.

5. The jump size between \(A_b\) and \(A_{b'}\) at \(x^* \in \partial A_b \cap \partial A_{b'}\) is defined to be
   \[
   \delta_{b,b'}(x^*) = \lim_{x \to x^*, x \in A_b} g_{b}(x) - \lim_{x \to x^*, x \in A_{b'}} g_{b'}(x).
   \]

   It is assumed that \(\delta_{b,b'}(x^*) \neq 0\) and they have the same sign, for any \(x^* \in \partial A_b \cap \partial A_{b'}\).

Estimation of \(m(x)\) has been studied using two different approaches. The first approach is to estimate the partition \(\{A_b; b = 1, ..., B\}\) and the corresponding JLCs, and then estimate \(m(x)\) using the conventional local smoothing procedures (e.g., kernel smoothing methods) within each sub-region \(A_b\) (Qiu and Yandell, 1997). By the second approach, the regression function \(m(x)\) is estimated by a one-sided kernel smoothing estimate, without explicit estimation of the JLCs (Qiu, 2009). However, optimizing the selection of the design points in a jump regression model has not been studied in these papers. The current article aims to develop a design selection strategy for jump regression analysis, primarily for applications such as the two motivating applications discussed in Section 1, but it is general enough to be applicable for other similar problems. To describe this design selection strategy, we first discuss the estimation of \(m(x)\) in a fixed design case and then derive the asymptotic bias and variance of the estimator in Section 3.1. We will relate the bias and variance of the estimator to the choice of the design points. We will then exploit this relationship to develop a new sequential experimental design that selects the design points to reduce the bias and variance of the estimated response surface in Section 3.2.

### 3.1. Model estimation in a fixed design case

Given observations \(Y_1, ..., Y_n\) at the design points \(\{x_1, ..., x_n\}\), we discuss nonparametric estimation of \(m(x)\), based on the one-sided local linear kernel smoothing approach (Qiu, 2009). In this article, we extend the approach with two modifications for our scientific applications. First, we assume that the design points are sparsely located in \(\mathcal{X}\), and their locations are non-uniformly distributed over \(\mathcal{X}\), as a result of optimizing the choice of design points in sequential design cases and other reasons. To accommodate such non-uniformly distributed design points, we use spatially varying kernel bandwidth instead of a constant bandwidth as used in Qiu (2009). Second, we extend the method from 2D cases (i.e., \(x\) has a dimension of two) to cases with two or more dimensions. In the scientific applications discussed in this article, the dimension is two. We tested the proposed approach in 3D cases using simulated datasets.

The one-sided kernel smoothing approach does not require explicit estimation of \(\{A_b\}\), and it gives a pointwise direct estimate of the regression function \(m(x)\) with the jumps in \(m(x)\) being accommodated automatically. At a given location \(x \in \mathcal{X}\), consider its neighborhood with the bandwidth \(h\):

\[
\mathcal{N}(x) = \{x' \in \mathcal{X} : d(x', x) \leq h\},
\]

where \(d(\cdot, \cdot)\) is the Euclidean distance. We seek a local estimate of \(m(x)\) using observed data in the neighborhood \(\mathcal{N}(x)\). In cases where the design points are uniformly distributed in \(\mathcal{X}\), a global bandwidth parameter is typically used as a function of the sample size \(n\). In this article, we allow the design points \(\{x_1, ..., x_n\}\) to be sampled from a non-uniform density \(f(x)\), due to the design selection procedure that we will discuss in the next section. To be more adaptive to the non-uniform density, we adopt spatially varying bandwidth parameters. Let \(h_n(x)\) denote the location-dependent bandwidth parameter, which is set to be the Euclidean distance from \(x\) to its \(k\)th nearest neighbor (k-NN) in \(\{x_1, ..., x_n\}\). The corresponding neighborhood of \(x\) is defined to be

\[
\mathcal{N}_n(x) := \{x' \in \mathcal{X} : d(x', x) \leq h_n(x)\}.
\]

Based on the existing literature on the k-NN density estimation (Wasserman, 2006), the k-NN bandwidth selection is asymptotically equivalent to selecting the bandwidth parameter to be inversely proportional to the density of the design points, i.e.,

\[
h_n(x) \propto \left(\frac{1}{nf(x)}\right)^{1/p}.
\]

Based on this asymptotic relationship, \(k\) should be chosen such that \(k = o(n)\) and \(k \to \infty\), as \(n \to \infty\) (Mack and Rosenblatt, 1979). Our choice in this article is \(k = \sqrt{n}\).

For the conventional local linear kernel smoother, a local estimate of \(m(x)\), for any location \(x \in \mathcal{X}\), is computed from all available observations in the local neighborhood \(\mathcal{N}_n(x)\). Different from the conventional local smoothing approaches, the one-sided local linear kernel estimate is
obtained using the observations in one of the two halves of \( N_n(x) \). The split of \( N_n(x) \) into two halves is done so that at least one of them is asymptotically on one side of the JLCs. To proceed, we first describe the conventional local linear kernel estimate and its error for estimating a jump regression function to motivate the need for the one-sided estimate. In the conventional local linear estimation, the function \( m(x) \) is locally approximated around a test location \( x \) by a linear model \( x + \beta^T (x_i + x) \), where \( \beta \) is the intercept and \( \beta \in \mathbb{R}^p \) is the slope. The slope and intercept are estimated so that the locally weighted sum of the residual squares is minimized:

\[
(\hat{\beta}, \hat{\beta}) = \arg \min_{\beta, \tilde{\beta}} \sum_{x_i \in N_n(x)} \left[ Y_i - \beta^T (x_i - x) \right]^2 K \left( \frac{x_i - x}{h_n(x)} \right),
\]

where \( K(u) \) is an isotropic kernel function with a unit-circle support \( \{ u \in \mathbb{R}^p : u^T u \leq 1 \} \). Then, the local estimate of \( m(x) \) at the location \( x_i = x \) is \( \hat{\beta} = \beta^T (x - x) \), which is simply \( \hat{\beta} \). Let \( \hat{m}(0)(x) \) denote the conventional local linear estimate of \( m(x) \).

The following theorem gives the asymptotic bias and variance of the estimate:

**Theorem 3.1.** Assume that \( g(x) \in C^2(\mathcal{X}) \) has a bounded second-order derivative, the kernel \( K \) is a Lipschitz-1 continuous and isotropic density function, and \( h_n(x) \) follows (2). For a given point \( x \in A_b \), if the projection of the point to the boundary set \( B \) is \( x_l \) and it is non-singular, i.e., \( x_l \in B \setminus S \), then there exists a unique pair of \( b \) and \( b' \) such that \( x_l \in \partial A_b \cap \partial A_{b'} \), and

\[
E[\hat{m}(0)(x)] - m(x) = \alpha \left( \frac{1}{n^{3/2} f(x)^{1/2}} \right) + (c_j + o_p(1)) \int_{\partial A_b} K(u) du,
\]

and

\[
\text{Var} \left[ \hat{m}(0)(x) | x_1, ..., x_2 \right] = \kappa \sigma^2 (1 + o_p(1)),
\]

where \( c_j = \delta_{b, b'}(x_l) \) is the jump magnitude at \( x_l \), \( \kappa \) is a constant that depends on the kernel function, and \( Q^{(b')} \) is the part of the kernel support that corresponds to \( A_{b'} \cap N_n(x) \).

The proof of the theorem is provided in the online supplementary material (Appendix A). From (4) and (5), the variance of the estimate is asymptotically a constant. The bias is significantly affected by \( d(x, x_l) \), the distance of the test point \( x \) to the nearest jump location curve. Please note that if \( d(x, x_l) \geq h_n(x) \), i.e., the test point is far away from the jump location curve, then \( A_{b'} \cap N_n(x) = \emptyset \) and consequently \( Q^{(b')} \) is an empty set. In such a case, the bias is simply \( \alpha \left( \frac{1}{n^{3/2} f(x)^{1/2}} \right) \), which is the same as the bias of the conventional local linear kernel estimate in a continuity region. However, when the distance goes below \( h_n(x) \), \( Q^{(b')} \) is non-empty, as illustrated in Figure 2(a). In such cases, the additional bias, \( c_j \int_{Q^{(b')}} K(u) du \), is generated. The additional bias is bounded above by

\[
c_j \int_{Q^{(b')}} K(u) du \leq c_j K \left( \frac{x - x_j}{h_n(x)} \right) \mathcal{L}(Q^{(b')}),
\]

where \( \mathcal{L}(\cdot) \) is the Lebesgue measure, and \( \mathcal{L}(Q^{(b')}) \) is

\[
O_p \left( \max \left\{ 0, 1 - \left( \frac{d(x_j, x)}{h_n(x)} \right)^p \right\} \right).
\]

Therefore, this part of the bias increases as \( d(x, x_l)/h_n(x) \) decreases, i.e., the test point approaches to the JLC.

To mitigate the bias increment, the local neighborhood \( N_n(x) \) is halved into \( N_n^{(1)}(x) \) and \( N_n^{(2)}(x) \), by a plane passing through \( x \) and being perpendicular to \( \beta(0) \), as illustrated in Figure 2(b), where \( \beta(0) \) is the solution to \( \beta \) in the conventional local linear estimation (3). According to Corollary 1 in Qiu (2009), \( \beta(0) \) is approximately perpendicular to the tangent plane of the jump location curve at \( x_l \) with some approximation error. Therefore, the cutting plane is approximately parallel to the tangent plane of the jump location curve, and either one of the two halves would be approximately on one side of the jump location curve. For example, in Figure 2(b), the test point \( x \) is in \( A_b \), and \( N_n^{(1)}(x) \) mostly belongs to \( A_b \) except for a small portion that corresponds to \( Q^{(b')} \) in the figure.
In each one-sided neighborhood \( \mathcal{N}_n^\ell(x) \), for \( l = 1, 2 \), we take the one-sided local linear kernel estimate of \( m \), denoted as \( \hat{m}_l(x) \), to be the solution of \( \alpha \) to the following optimization problem:

\[
(\hat{m}_l(x), \hat{b}_l(x)) = \arg \min_{x, \hat{b}} \sum_{x \in \mathcal{N}_n^\ell(x)} [Y - z - \beta^T(x_i - x)]^2 K \left( \frac{x_i - x}{h(x)} \right).
\]  

(7)

The final estimate of \( m(x) \) is chosen to be one of \( \hat{m}_1(x) \) or \( \hat{m}_2(x) \), and the choice depends on their estimation errors. The bias and variance of the two one-sided estimates are given in Theorem 3.2. The proof of Theorem 3.2 is similar to that of Theorem 3.1.

**Theorem 3.2.** Under the same conditions stated in Theorem 3.1, we have

\[
E[\hat{m}_1(x)] - m(x) = o_p \left( \frac{1}{n^{3/2}f(x)^{3/2}} \right)
\]

\[
+ (2c_j + o_p(1)) \int_{Q^{2g'}} K(u) du,
\]

(8)

and

\[
E[\hat{m}_2(x)] - m(x) = o_p \left( \frac{1}{n^{3/2}f(x)^{3/2}} \right)
\]

\[
+ (2c_j + o_p(1)) \int_{Q^{2g'}} K(u) du,
\]

(9)

where \( Q^{2g'} \) is the part of the kernel support that corresponds to \( A_U \cap \mathcal{N}_n^\ell(x) \).

By the above theorem, the variances of the two one-sided estimates are asymptotically the same. Therefore, the mean squared errors of the estimates are largely influenced by their respective bias terms. The major parts of the asymptotic biases are \( 2c_j \int_{Q^{2g'}} K(u) du \) and \( 2c_j \int_{Q^{2g'}} K(u) du \). Since \( Q^{1g'} \cup Q^{2g'} = Q^{2g'} \), the two terms can be written as

\[
2c_j \int_{Q^{2g'}} K(u) du = a_1 c_j \int_{Q^{2g'}} K(u) du
\]

and

\[
2c_j \int_{Q^{2g'}} K(u) du = (2 - a_1) c_j \int_{Q^{2g'}} K(u) du,
\]

for a constant \( a_1 \in [0, 2] \). The smaller value of the two terms is bounded above by

\[
2c_j \min \left\{ \int_{Q^{2g'}} K(u) du, \frac{1}{a_1} c_j \int_{Q^{2g'}} K(u) du \right\}
\]

\[
\leq c_j K \left( \frac{x - x_i}{h_i(x)} \right) Op \left( \max \left\{ 0, 1 - \left( \frac{d(x_i, x)}{h_i(x)} \right)^p \right\} \right) \min\{a_1, 2 - a_1\}.
\]  

(10)

The last term, \( \min\{a_1, 2 - a_1\} \), depends only on \( \hat{b}_l(x) \). When \( \hat{b}_l(x) \) along the tangent plane at \( x_i \), the value of \( \min\{a_1, 2 - a_1\} \) is approximately at its maximum, a value of one. When the direction of \( \hat{b}_l(x) \) is perpendicular to the tangent plane, this value is zero. As mentioned earlier, it has been confirmed that \( \hat{b}_l(x) \) is asymptotically perpendicular to the tangent plane (Qiu, 2009). Thus, \( \min\{a_1, 2 - a_1\} \) is approximately zero.

The bias terms cannot be numerically evaluated since \( Q^{1g'} \) and \( Q^{2g'} \) are unknown. To make a choice between \( \hat{m}_1(x) \) and \( \hat{m}_2(x) \), the following weighted residual mean errors are considered:

\[
err_l(x) = \int_{x \in \mathcal{X}} \frac{[Y - \hat{m}_l(x) - \hat{b}_l(x)]^2 K(\frac{x_i - x}{h_i(x)})}{\sum_{x \in \mathcal{N}_n^\ell(x)} K(\frac{x_i - x}{h_i(x)})} dx.
\]

When \( err_1(x) < err_2(x) \), \( \hat{m}_1(x) \) is chosen; and \( \hat{m}_2(x) \) is chosen otherwise.

### 3.2. Proposed method for sequential design selection

This section describes our proposed method for a sequential selection of design points, which selects the design points in multiple stages. The first stage serves as a seed stage, and the design points in the first stage are randomly sampled from a uniform distribution or selected by Latin Hypercube Sampling (LHS). In all of our numerical examples, we used LHS in the first stage. Each of the subsequent stages is described as follows. Suppose that there are \( n \) design points selected up to the previous stage, and we describe how \( b \) additional design points are selected in the current stage. Let \( f_1 \) denote the unknown density of the \( n \) design points from the previous stages, and let \( f_{21} \) represent the sampling density used to select the \( b \) design points in the current stage. If \( f \) was a "desirable" joint density of the \( n \) design points and the \( b \) additional design points, then the sampling density for the current stage's design points should be the following conditional density:

\[
f_{21}(x) = \frac{f(x)}{f_1(x)}.
\]

Intuitively, \( f \) should be chosen to minimize the integrated square loss:

\[
\int_{x \in \mathcal{X}} E[|m(x) - \hat{m}(x)|^2] dx.
\]

Note that the square loss \( E[|m(x) - \hat{m}(x)|^2] \) can be decomposed into the squared-bias and the variance of \( \hat{m}(x) \). Based on Theorem 3.2, the variance of the jump regression estimate defined in Section 3.1 is approximately a constant, and the square loss is largely influenced by the squared-bias term. We will develop our sequential design strategy to balance the bias and the integrated square loss function. To develop the idea more formally, please note that the bias can be as small as \( o_p \left( \frac{1}{n^{3/2}f(x)^{3/2}} \right) \) when the test location is far away from the jump location curve in the sense that \( d(x, x_i) \geq h_i(x) \). If \( d(x, x_i) < h_i(x) \), there is an additional bias of size \( 2c_j \min \left\{ \int_{Q^{2g'}} K(u) du, \frac{1}{a_1} c_j \int_{Q^{2g'}} K(u) du \right\} \). By (10), this part of the bias is bounded above by

\[
2c_j \min \left\{ \int_{Q^{2g'}} K(u) du, \frac{1}{a_1} c_j \int_{Q^{2g'}} K(u) du \right\} \min\{a_1, 2 - a_1\}.
\]  

(11)
which goes to zero as \( \frac{d(x_j, x)}{h_n^2} \) increases or \( d(x_j, x)f(x) \) increases. To balance the bias over \( x \) and minimize the integrated square loss function, the desirable sampling density should be

\[
f(x) \propto \frac{1}{d(x_j, x)}.
\]

We hope that collectively the \( n + b \) design points have higher densities at places near the JLC (i.e., places with small \( d(x_j, x) \)). Certainly, we do not know where the JLCs are located in practice, so we do not know the distance \( d(x_j, x) \). However, the distance can be roughly located using the observations of the regression function at the \( n \) design points selected in the previous stages. It is easy to show that the following statistic

\[
\left[ \hat{m}_1(x) - \hat{m}_2(x) \right]^2,
\]

so we use it as a jump detection statistic. Based on the jump detection statistic, we propose a desirable joint density \( f \) to be

\[
f(x) = C \exp \left\{ \gamma \left[ \hat{m}_1(x) - \hat{m}_2(x) \right]^2 \right\}, \quad x \in \mathcal{X},
\]

where \( C > 0 \) is a normalization constant, and the coefficient \( \gamma \) controls the exploration vs exploitation trade-off. We chose \( \gamma = 1/\sigma^2 \), where \( \sigma^2 \) is the noise variance. With that choice, the quantity \( \gamma (\hat{m}_1 - \hat{m}_2)^2 \) is approximately the square of the jump magnitude relative to the noise variance. For a larger \( \sigma^2 \), this sampling function seeks more exploration, and for a smaller \( \sigma^2 \), more exploitation is sought. The noise standard deviation \( \sigma \) is estimated using the median absolute deviation.

As \( \mathcal{X} \) is bounded and the estimates \( \hat{m}_1 \) are bounded, \( C \) is well defined. From (11), the sampling density for the \( b \) new design points should be

\[
f_{2|1}(x) = \frac{f(x)}{f_1(x)} \approx \frac{C \exp \left\{ (\hat{m}_1(x) - \hat{m}_2(x))^2 \right\}}{1 / n \sum_{i=1}^n K \left( \frac{x-x_i}{h_n} \right)},
\]

where the approximation comes from the standard kernel density estimation of \( f_1(x) \), and \( h \) is a non-adaptive kernel bandwidth parameter that depends on the sample size \( n \). Sampling from the complex density (14) can be performed by the Metropolis–Hastings Algorithm. We can limit the sampling locations to the ones sampled from the uniform distribution over the regression domain for more computational feasibility. For each of the possible sampling locations, we can compute \( f_{2|1}(x) \) up to a normalizing constant. The computed values are normalized so that the summation of all the computed values is equal to one. The normalized values will serve as the probability mass function (pmf) defined on a finite number of the possible sampling locations. Then \( b \) samples will be drawn randomly from that pmf as the \( b \) new design points for the next stage.

\[\text{Normalized Values} = \frac{\text{Computed Values}}{\text{Sum of all Computed Values}}\]

### 4. Simulation study: 2D and 3D domains

For the initial validation of the proposed method, we performed a simulation study with three synthetic datasets. Figure 3 shows the underlying noise-free regression functions for the first two synthetic datasets defined on the 2D domain \([0,200]^2\), and the regression function for the third dataset defined on the 3D domain \([0,50]^3\). The underlying noise-free regression functions are in the mixture form:

\[
m(x) = g_0(x) - 0.3I_{A_b}(x),
\]

where \( g_0 \) is continuous on \( \mathcal{X} \), and \( A_b \subset \mathcal{X} \) represents the subregion with a different intensity level. Then, the regression function is continuous except at the boundary \( \partial A_b \subset \mathcal{X} \).

For the first two datasets:

\[
g_0(x) = \sin \left( \frac{x_1}{20} \right) \times \cos \left( \frac{x_2}{20} \right),
\]

where \( x_1 \) and \( x_2 \) are the first and second elements of the input vector \( x \) respectively. For the third dataset, we used

\[
g_0(x) = \sin \left( \frac{x_1}{5} \right) \times \cos \left( \frac{x_2}{5} \right) \times \sin \left( \frac{x_3}{5} \right),
\]

where \( x_1, x_2 \) and \( x_3 \) are the first, second and third elements of \( x \) respectively. In Figure 3, the set \( A_b \) is shown as the dark regions. We then added i.i.d Gaussian noise from \( \mathcal{N}(0, \sigma^2) \) to \( m(x) \) to obtain observed data.

For each dataset, \( n \) design points in total were selected, using our proposed sequential adaptive approach described in Section 3.2. We varied the number \( n \) and the number of design points chosen in each stage, denoted by \( b \). Since the domain sizes differ in the three datasets, which are 200 x 200 in the first two datasets and 50 x 50 x 50 in the last dataset, the experimental settings are denoted in terms of the percents of the domain sizes. The number \( n \) varies over 2.5%, 3.75%, 5.00%, 6.25%, 7.50%, 8.75% and 10% of the domain size. The number \( b \) varies over 0.125%, 0.25%, 0.625%, and 1.25% of the domain size. We also varied the noise level \( \sigma \) over 0.1, 0.2, 0.4, 0.6, 0.8, 1 to study different Signal-to-Noise Ratio (SNR) cases. In the case of \( \sigma = 1 \), the noise level is

\[\text{SNR} = \frac{\text{Signal}}{\text{Noise}}\]

Figure 3. Three synthetic datasets: (a) 2D-others, (b) 2D-star, (c) 3D-donut.
equal to the maximum signal intensity. Combining the three parameter values, we had a total of 168 different simulation scenarios, and each scenario was run for 20 replications. Figure 4 illustrates the selected design points obtained in the case with $n = 10\%$ and $b = 1.25\%$. The design points selected for the first stage are seed locations, selected by LHS, and the design points selected in the subsequent stages are more concentrated around the jump boundaries and some intensity transitioning areas.

After all the design points were selected, the noisy observations of the regression function at the design points served as training data to estimate the regression function. Test locations were randomly sampled from a uniform distribution over the regression domain, excluding the ones overlapped with the training data, and the estimate of the regression function was computed for each of the test locations, using the procedure in Section 3.1. The estimates were compared to the corresponding true regression function values (serving as the ground truth) at the test locations to evaluate the mean square errors. We used two Mean Square Error (MSE) metrics: MSE near jump location curves and MSE in the continuity regions, defined to be

\[
\text{J-MSE} = \frac{1}{|J[h]|^0} \sum_{(x,y) \in J[h]} (\hat{m}(x) - m(x))^2
\]

\[
\text{C-MSE} = \frac{1}{|J[h]|^c} \sum_{(x,y) \in J[h]^c} (\hat{m}(x) - m(x))^2,
\]

where $\hat{m}(x)$ is the jump regression estimate, $J[h]$ is the set of the test locations whose distance from the closest JLC is less than or equal to $h$, and $J[h]^c$ is the complement of $J[h]$; $h$ is fixed to be six, which is about twice of the average distance between two neighboring pixels.

4.1. Effect of the tuning parameters, $n$, $b$ and $\sigma^2$

We first evaluate how the proposed approach performs under various experimental settings. Figure 5 shows the changes in J-MSE and C-MSE for different settings specified by the total number of selected design points, denoted by $n$, and the number of design points selected per stage, denoted by $b$. The per-stage selection size $b$ determines the number of stages for a fixed $n$. According to Figure 5, the per-stage selection size is not the major factor that affects J-MSE and C-MSE. For the first two test datasets with 2D domains, the per-stage selection size does not make any significant difference in both J-MSE and C-MSE. For the last test dataset with 3D domain, the J-MSE tends to be lower for a smaller $b$ and the C-MSE tends to be lower for a larger $b$.

Both accuracy measures are more significantly affected by $n$. Based on the results, we would recommend setting $n$ to meet a required level of accuracy and choose a large $b$ for a computational gain. The number of stages to get $n$ design points is proportional to $n/b$. If $b$ is too small, many stages would be needed, and more frequent computations to update the sampling density function are needed. Therefore, the total computation time would increase. For the remainder of our numerical experiments, we used $b = 1.25\%$, the largest value we tried. We also look at the two performance measures for different noise levels of the observed data. We can see clear downward trends in both J-MSE and C-MSE as the noise level decreases or SNR increases. More details can be found in the online supplementary material (Appendix B).
4.2. Comparison with four benchmarks

We compared the performance of the proposed approach with four benchmark methods. The proposed approach is denoted as JuMp Planner (JMP). The first benchmark is randomly sampling from a uniform density (RAND), the second approach is sampling with LHS. The third approach is sampling from a density proportional to the Weighted Residual Mean Square (WRMS) error of the conventional local linear kernel smoother:

$$\text{WRMS-C}(x) = \frac{\sum_{x_i \in \mathcal{N}_a(x)} Y_i - \hat{\beta}^T (x_i - x)^2 K \left( \frac{x_i - x}{h_{\beta}(x)} \right)}{\sum_{x_i \in \mathcal{N}_a(x)} K \left( \frac{x_i - x}{h_{\beta}(x)} \right)}.$$
where \( \hat{z} \) and \( \hat{\beta} \) are the optimal solutions of problem (3). The last benchmark is sampling from a density proportional to the WRMS error of the jump regression model (Qiu, 2004):

\[
\text{WRMS-J}(x) = \min \{ \text{err}^{(1)}(x), \text{err}^{(2)}(x) \}.
\]

In this comparison, we fixed \( b = 1.25\% \) and \( n = 10\% \), as different choices of \( b \) and \( n \) did not make much difference in the comparison results. Figure 6 shows SNR versus the averages of the two MSE metrics over 20 replicated simulation runs. From the figure, it can be seen that the C-MSE...
values computed over the continuity regions do not depend on the choice of the design selection method. However, the J-MSE values computed near the jump locations differ significantly among different methods. The major findings regarding J-MSE are summarized below:

- **Low Noise Case, \(\sigma = 0.1\) or SNR = 2**: The three error-based methods, JMP, WRMS-J and WRMS-C, significantly outperformed the two random sampling methods, RAND and LHS.

- **High Noise Case, \(\sigma = 1, \text{or} \text{SNR} = 0\)** (i.e., maximum intensity of the regression function is equal to \(\sigma\)): All methods are comparable, which is not surprising. When the noise level is comparable to the maximum signal intensity, the error-guided methods cannot cleanly distinguish signals from noise. In such cases, the three adaptive selection strategies work similarly to the two random sampling methods.

- **Medium Noise Cases, \(0.1 < \sigma < 0.8\)**: The methods JMP and WRMS-J outperform the method WRMS-C. Namely, the two jump regression-based approaches are superior to the conventional local smoothing approach around the jump regions. Furthermore, the method JMP is better than WRMS-J in the first two examples with 2D data, and comparable to WRMS-J in the third example with 3D data.

We also present the variabilities of the two MSE metrics based on 20 replicated simulation runs. Figure 7 shows the standard deviations of J-MSE and C-MSE values at different noise levels. The overall variabilities of the compared methods increase as \(\sigma\) increases or SNR decreases. Among the five methods, the proposed approach has the lowest variability in most cases considered. Thus, a low variability is another advantage of the proposed approach.

### 4.3. Comparison with existing adaptive sampling strategies

In this part, we compare the proposed method with two existing adaptive design strategies for tree regression models, including Taddy’s dynamic tree model (Taddy et al., 2011, dynaTree) and the Bayesian Treed GP (Gramacy and Lee, 2009, btgp). For this comparison, we used R-libraries dynaTree and tgp. A major issue that makes the comparison difficult is that the computation for the dynaTree and btgp is too expensive for a large-size problem. For example, when they are applied to the first simulated scenario considered in the previous section, the dynamic tree model took 433.65 seconds to sample 4000 design points, and the btgp method took 1505 seconds for the first two stages of samplings alone.

Considering the heavy computation involved in these existing approaches, we consider cases with a small domain size and a small number of sampling size to make a comparison. In the toy example, the true regression function is defined in the small 2D domain \([0, 2]^2\) in the mixture form:

\[
m(x) = g_0(x) + 3L_{A_0}(x),
\]

where \(g_0(x) = \sin \left(\frac{x_1}{2}\right) \times \cos \left(\frac{x_2}{4}\right)\) is a continuous function, and \(A_0 = \{x_2 = 1/x_1\}\). To estimate the function, the noisy observations of the function at 125 locations are considered, and the noisy observations are generated by adding Gaussian noise from \(N(0, 0.4^2)\) to \(m(x)\). The 125 design points are selected by our proposed sampling approach, the ALM sampling function with dynaTree, and the ALM sampling function with btgp, respectively. For all three methods, the 50 initial seed locations are selected by the LHS, and the remaining 75 locations are selected sequentially over 75 sampling stages, one per stage.

Figure 8 compares the design points selected by the three adaptive design strategies and also shows the regression functions estimated by the respective regression approaches (i.e., proposed jump regression, dynamic tree model and Bayesian treed GP) with the selected design points. As shown in Figure 8(b), the proposed approach places more design points around the jump boundary, so the resulting regression estimate is close to the ground truth shown in panel (a). The dynamic tree model partitions the input domain into five rectangular regions. Please note that the tree model fits a linear leaf model for each sub region. The regression estimate with five linear leaves appears to be too simple to represent the true regression surface with a curvy jump boundary, as shown in Figure 8(c). The design points selected by this approach are mostly located around the corners of the rectangular sub regions. The btgp model is a more flexible model than the dynamic tree model with GP leaves. However, in this numerical example, the Bayesian tree has only one GP leaf, so one stationary GP model is used to describe the regression function. Therefore, the corresponding regression estimate has quite a large bias around the jump boundary, as shown in Figure 8(d). The design points selected by this approach are similar to the ones selected by LHS.

Table 1 makes a quantitative comparison among the three methods in terms of J-MSE, C-MSE and the computing time. The proposed sampling approach is superior in all three metrics, because it is based on a more flexible regression modeling, obtains better sampling locations, and thus has better MSES in both continuous and discontinuous regions. This study also confirms the computational burden of the tree-based approaches, which makes them inappropriate for applications, such as the first motivating example about STEM imaging discussed in Sections 1 and 5.

### 5. Real data study: Adaptive microscope imaging for accelerating the imaging speed

Accelerating the STEM imaging speed would open unprecedented opportunities in studying important material processes. One promising method to accelerate the speed is to scan material specimens in a reduced set of spatial locations. Here we illustrate the use of our proposed sequential adaptive design strategy to select only a targeted, partial set of scan locations. Two important factors need to be considered: accuracy of estimating the specimen image at unobserved locations (i.e., regression accuracy) and the computation time spent to calculate design selection statistics and sample design points. We do not want the required computation time to surpass the full scan time to accelerate the total imaging acquisition time. This requirement on computing
time makes many Bayesian tree-based sequential design approaches ineligible for this application.

5.1. Application details

In this application, the total sample size $n$ is set to be 10% of the pixel number of the full scan, considering the accuracy requirement. The 10% partial scan can provide good accuracy for estimating the underlying images based on prior numerical trials. Lowering the sample size could lead to the loss of many sharp features of the material images, and increasing the sample size would increase the imaging time. In STEM imaging, the time to scan one pixel is referred to as a pixel dwell time, which is about 10 to 40 microseconds. To achieve good quality pixel measurements,
40 microseconds of pixel dwell time is applied. The total imaging time is approximately the number of pixels to scan, multiplied by the pixel dwell time. For example, scanning a $587 \times 484$ imaging area would take $587 \times 484 \times 40\mu$s, equal to 10.9 seconds in total. If we only select 10% of the imaging pixels for a partial scan, then the physical scanning time would be only 1.09 seconds, i.e., 10% of the full scan time.

Selecting design points in multiple stages would require a significant computing time, as the sampling density needs to be recalculated every stage based on the results in all previous stages. The total computing time increases as the number of stages, $M$, increases. Therefore, the number $M$ in the sequential design selection scheme should be carefully selected, considering the computing time and image reconstruction accuracy with the selected samples. We first performed some initial experiments to choose the appropriate number for $M$. Table 2 shows the computing times and reconstruction accuracy for several different values of $M$ when 10% of a $587 \times 484$ test image is sub-sampled using the proposed approach.

From the table, it can be seen that the computing time increases and the reconstruction error decreases as $M$ increases. However, the reduction in the reconstruction error is saturated after $M = 4$. We have similar outcomes for other test images. Therefore, we choose $M = 4$ to balance the accuracy and the computing time in this application. If the number of stages is $M = 4$, the total computing time of the partial scan would include 1.09 seconds of the physical scan time plus 2.5678 seconds of the computing time, which is about three times shorter than the time for the full scan. The imaging accelerating factor would be three in such a case. We will use $M = 4$ for the remainder of this section, which corresponds to $b = 2.5\%$.

### 5.2. Applications to STEM imaging under various different conditions

To quantitatively evaluate our approach compared with the standard STEM imaging techniques, we first obtained complete imaging scans for 11 different specimens (Figure 9) to serve as ground truth. Their noise levels characterize these microscope images. The noise levels of the images are estimated as follows. We first used the jump regression estimates of the images for denoising. The noise variances are estimated by the mean squared differences of the estimated regression surfaces and the corresponding original images. The 11 images have $587 \times 484$, $587 \times 465$, $611 \times 474$, $...
In each of the 11 cases, we also achieved the partial scan using our proposed approach. The number \( n \) selected equals 10% of the total raster location number, and the locations were selected sequentially in six stages. The subset of the raster locations and the corresponding pixel measurements were used to estimate the pixel measurements at the other unselected raster locations. The estimates were compared to the corresponding values of the full raster image, and the two performance metrics, J-MSE and C-MSE, were computed. The evaluation of J-MSE requires \( J_B(h) \), which was estimated. We first applied an image segmentation algorithm to identify the outlines of black regions, and the results of the image segmentation algorithm were manually corrected for better accuracy, and then \( J_B(h) \) is estimated accordingly. Samplings from the uniform density (RAND), WRMS-C, WRMS-J and LHS were used as benchmarks here.

Figure 10 shows the comparison of the related methods in terms of the two performance metrics. We used \( n = 3\% \) for the illustration because the red dots are too dense to show the results effectively otherwise. A few key findings are summarized below:

- For all test images, the Root C-MSE values for different design selection strategies are comparable and close to the noise level. This is consistent with what we found in the simulation study.
- MG1 through MG4 (low noise and high ratio of jump location boundary pixels): Sampling from WRMS-J and the proposed design selection strategy are significantly better than the other methods. This is also consistent with the findings from the simulated studies.
- MG5 through MG7 (medium noise): The proposed design selection approach is better than all the other methods with significant margins, while sampling from WRMS-C is not much better than Random Sampling.
- MG8 (high noise and many tiny foregrounds): Sampling from WRMS-J and the proposed design selection strategy are significantly better than sampling from WRMS-C and Random Sampling.
- MG9 through MG11 (very high noise): All methods perform similarly. The proposed strategy is based on the jump detection statistic, which is almost uniform when the noise level is comparable to the jump size \( c_J \), so the strategy becomes similar to the uniform sampling

| No. Stages (M) | \( M = 1 \) | \( M = 2 \) | \( M = 3 \) | \( M = 4 \) | \( M = 5 \) | \( M = 10 \) |
|----------------|--------------|-------------|-------------|-------------|-------------|-------------|
| Computing Time (in seconds) | 0.0954 | 0.9402 | 1.7832 | 2.5678 | 3.4383 | 7.5855 |
| Reconstruction Error | 0.0237 | 0.0205 | 0.0197 | 0.0195 | 0.0195 | 0.0194 |

Figure 9. Full raster scanned microscope images. Each image was labeled with image number and \( \sigma \) (in parenthesis), where \( \sigma \) is the noise standard deviation when the image intensity is normalized so that its maximum is one.
strategy as shown in Figure 2 of the online supplementary material (Appendix C).

In summary, the proposed approach is very promising in compressive and adaptive imaging for accelerating the image scan speed in STEM unless the level of image noise is comparable to the intensity jumps at edges.

6. Real data study: Experimental campaign for predicting carbon nanotube growth

This section presents the application of the proposed approach to another motivating example of this article, a problem of optimizing an experimental campaign for predicting the response variable of a chemical experiment under a given experimental condition when the response jumps around certain characteristic boundaries.

6.1. Scientific background and significance

We use a chemical experiment of carbon nanotube (CNT) growth as a motivating example. CNTs are tubes made of carbon atoms with nano-scale diameters. The nanotubes exhibit exceptional tensile strengths and great thermal/electrical conductivity, which are being applied to many practical applications. They are chemically synthesized using a Chemical Vapor Deposition (CVD) process. We are interested in understanding how the reaction conditions of the chemical process affect CNT growth. The dependent variable of interest (i.e., the response variable) is the resultant amount of CNT growth under a given reaction condition, and the input variables describe the reaction condition. Among many process parameters describing the reaction condition, the reaction temperature and the composition of chemical reactants greatly affect the growth outcomes, which are the two experimental inputs. The first chemical reactant is C2H4, which is a catalyst to promote the growth reaction. The second reactant is CO2, which suppresses the growth reaction. When the concentration ratio of the two chemicals is below a certain threshold, the CNT growth is kept low to almost a zero level. However, the amount suddenly jumps to a certain level right above that threshold. The observed jump behaviors in the closed-loop CNT growth directly result from a catalyst phase transition, and the underlying physics is discussed in greater detail in our upcoming publication (Carpena-Núñez et al., 2020).

Estimating the response surface embedding jumps would require many experiments if one uses a uniform design such as a space-filling design. That is mainly due to the presence of sharp jumps in the response surface and locating the sharp jumps precisely is possible only when the design points are uniformly dense over the design space unless a non-uniform design is adopted. From a past experimental campaign of the same kind done at Air Force Research Lab (AFRL), about 70 design points were selected manually by a human operator, and many of the design points were located in not unuseful zero-flat growth regions. To make the experimental campaign more efficient, we applied our proposed design selection strategy to select a design for estimating the response surface with increased fidelity in locating the jump structures. The accurate estimation of the response surface and the embedded jumps would guide practitioners to design CVD processes for good CNT yields.

6.2. Application details

The sequential process is implemented by AFRL using a research robot, Autonomous Research System (ARES), that performs CNT growth experiments. A detailed description of the growth experiments can be found in our previous works (Rao et al., 2012; Nikolaev et al., 2014; Nikolaev et al., 2016). We limit the design space to practical ranges of the two input variables. The suitable range for the concentration ratio of the two chemical reactants is 0 to 6.7 in the log scale or 1 to 800. The ratio below zero cannot expect any growth because there is more growth suppressor (CO2) than catalyst. The ratio 800 is regarded as almost pure catalysts, so the further increase of the ratio would not be more effective. The reaction
temperature ranges from 600 to 1100 °C. The temperature below 550 °C is too low to induce the growth of CNTs, and the temperature above 1200 °C is difficult to apply given a heat source and the melting temperature of supporting materials. The design space would be $[0, 6.7] \times [600, 1100]$ of the log-ratio and temperature. We use the proposed multi-stage sequential design approach to explore the response surface over the design space. The first stage is the seed experiment, and the experimental design of the first stage is hand-picked by an expert. For running experiments efficiently, five distinct values of the concentration ratio are tried, which are 1, 10, 100, 400, and 680 or 0, 2.3, 4.6, 6.0, and 6.5 in the log scale. For each of the five values, five to seven reaction temperatures are tried, and the reaction temperatures are hand-picked from the temperature range where jumps in growth are expected based on some prior engineering knowledge. In total, 31 design points are selected for the seed stage. In the second stage and thereafter, the design points are chosen by our proposed approach. Given experimental costs, each stage cannot perform too many experiments, so we run 20 experiments per stage. The stages continue until we have a satisfactory outcome. Therefore, the total selection size $n$ is adaptively chosen.

### 6.3. Results

Figure 11(a) shows the response surface estimated with the experimental outcomes at the 31 design points of the first stage, whereas Figure 11(b) shows the jump detection statistic (Equation (12)) estimated with the first-stage sample. The yellow band with a quite wide bandwidth in the figure is the potential jump region. The wide bandwidth implies that the region of jumps in nanotube growth is not narrowed down, so we need to do more experiments to narrow it down. Based on the statistics and the corresponding sampling density $f_{j1}$, the second-stage samples are taken as shown in Figure 11(b). The samples are mostly from the yellow band. After the second-stage is completed, the response surface is re-estimated as shown in Figure 11(c), and the corresponding jump detection statistic is estimated as shown in Figure 11(d). In the figure, we observe a narrow region where the jump detection statistic has much higher values than the other region. This narrow region corresponds to where the response surface jumps around. As the jump region is narrowed down enough, we decided to stop the design selection. In total, we took 51 design points, which is a tiny number compared with more than several hundred
design points necessary to narrow down the jump region following the uniform design of experiments.

7. Conclusion

We proposed a novel adaptive design strategy (cf., (14)) for sequential selection of design points in JLCs. The proposed method originated from our asymptotic error analysis of the jump regression estimate, which showed that placing more design points around the JLCs would give a faster decay of the integrated mean square regression error. Therefore, the proposed sampling function has a large density around the jump location curves. We applied the proposed strategy to two materials science applications: the compressive material imaging problem, in which sub-sampled images are used to reconstruct full images and the design selection to accelerate a materials discovery. The outcomes are promising. We have shown that STEM imaging can be accelerated at least 10 times faster, while sharp image features are preserved unless the image noise level is comparable to or higher than the image contrast. We also showed from the second example that experimental campaigns for materials discovery in CNTs can be accelerated by using our proposed approach.

Funding

We acknowledge support for this work from the AFOSR (FA9550-18-1-0144), NSF (DMS-1914639) and Oak Ridge National Laboratory (4000152630).

ORCID

Chiwoo Park ORCID http://orcid.org/0000-0002-2463-8901

Data availability statement

The data that support the findings of this study are available from the corresponding author, Chiwoo Park (cpark5@fsu.edu), upon request.

Notes on contributors

Chiwoo Park received his BS in industrial engineering at Seoul National University and a PhD degree in industrial engineering at Texas A&M University in 2011. He is currently an Associate Professor in the Department of Industrial and Manufacturing Engineering at Florida State University and a principal investigator at High Performance Materials Institute. His main research interest lies in machine learning and data science, focusing on object data (such as image, shape, motion, function and directional data) and surrogate modeling.

Peihua Qiu is the founding chair of the Department of Biostatistics at the University of Florida. He was previously a professor in the School of Statistics at the University of Minnesota. He was the editor of Technometrics, an elected fellow of the American Statistical Association and the Institute of Mathematical Statistics, and an elected member of the International Statistical Institute. His research focuses on jump regression analysis, medical image analysis, statistical methods for monitoring processes, and patient survival data analysis.

Jennifer Carpena-Náñez is a chemical physicist at Air Force Research Laboratory, Materials and Manufacturing Directorate at Wright-Patterson Air Force Base, OH. She earned a PhD degree in Physics from University of Puerto Rico. Her main research is the studies of carbon nanotubes: synthesis, characterization and structures.

Rahul Rao is a research physical scientist at Air Force Research Laboratory, Materials and Manufacturing Directorate at Wright-Patterson Air Force Base, OH. He has an MS in Materials Science and a PhD in Physics from Clemson University. The main focus of his research at AFRL was to conduct in-situ Raman spectroscopy studies during carbon nanotube growth, as well as studying novel catalyst systems, with the ultimate aim of understanding the fundamental principles behind carbon nanotube growth.

Michael Susner earned BS in Chemistry (2005) from Michigan State University and received MS (2009) and PhD (2012) in Materials Science and Engineering from The Ohio State University. From 2014-2016 he was a Postdoctoral Research Fellow in the Correlated Electron Materials Research Group at Oak Ridge National Laboratory. He joined the Air Force Research Laboratory in 2017 as a NRC Fellow and worked in the Soft Matter Materials Branch in the Materials and Manufacturing Directorate as a UES Research Scientist from 2019–2020. He is currently a research materials engineer in the Photonic Materials Branch. He is interested in establishing structure-property correlations in functional materials, i.e., those evincing magnetic, ferroelectric, and superconducting behaviors, particularly in single crystal materials.

Benji Maruyama is a Principal Materials Research Engineer in the Air Force Research Laboratory, Materials & Manufacturing Directorate, Autonomous Materials Lead & ACT3 (Autonomous Capabilities Team 3) Liaison lead. His focus area is the synthesis and processing science of carbon nanotubes using ARES which is the first fully Autonomous Research (ARES) Robot for materials. Dr. Maruyama’s interests include the research process itself, for which he promotes the ‘Moore’s Law for the speed of research”. He is also the point of contact for carbon materials for the AFRL Materials & Manufacturing Directorate. His materials interests include carbon nanomaterials, energy storage, flexible-hybrid materials and processes, field emission, carbon, polymer and metal matrix composites, imaging of complex 3D microstructures and AI/Machine Learning. He is currently involved in the study of the origins of chiral growth for carbon nanotubes, defect engineering for low dimensional materials, catalysis and autonomous experimentation.

References

Arias-Castro, E., Candes, E.J. and Davenport, M.A. (2013) On the fundamental limits of adaptive sensing. *IEEE Transactions on Information Theory, 59*(1), 472–481.

Bull, A.D. (2013) Spatially-adaptive sensing in nonparametric regression. *The Annals of Statistics, 41*(1), 41–62.

Carpena-Núñez, J., Nikolaev, P., Susner, M., Rao, R., Gorisse, S., Park, C. and Maruyama, B. (2020) Mapping carbon nanotube catalyst phase transitions using jump regression. Unpublished manuscript.

Chaudhuri, P. and Mykland, P.A. (1993) Nonlinear experiments: Optimal design and inference based on likelihood. *Journal of the American Statistical Association, 88*(422), 538–546.

Chernoff, H. (1972) Sequential Analysis and Optimal Design. Society for Industrial and Applied Mathematics, Philadelphia PA.

Cohn, D.A., Ghahramani, Z. and Jordan, M.I. (1996) Active learning with statistical models. *Journal of Artificial Intelligence Research, 4*(1), 129–145.

Cortes, C., DeSalvo, G., Gentile, C., Mohri, M. and Zhang, N. (2019) Region-based active learning, in *Proceedings of the Twenty-Second International Conference on Artificial Intelligence and Statistics*, Volume 89, pp. 2801–2809, PMLR, Okinawa, Japan.

Dror, H.A. and Steinberg, D.M. (2008) Sequential experimental designs for generalized linear models. *Journal of the American Statistical Association, 103*(481), 288–298.
Goetz, J., Tewari, A. and Zimmerman, P. (2018) Active learning for non-parametric regression using purely random trees, in *Advances in Neural Information Processing Systems*, Proceedings of 32nd Conference on Neural Information Processing Systems, Volume 31, pp. 2542–2551, MIT Press, Montréal, Canada.

Gramacy, R.B. and Lee, H.K. (2009) Adaptive design and analysis of supercomputer experiments. *Technometrics*, 51(2), 130–145.

Hoang, T.N., Low, B.K.H., Jailet, P. and Kankanahalli, M. (2014) Nonmyopic ε-Bayes-optimal active learning of Gaussian processes, in *Proceedings of the 31st International Conference on Machine Learning*, Volume 32, pp. 739–747, PMLR, Beijing, China.

Kim, H.-M., Mallick, B.K. and Holmes, C.C. (2005) Analyzing nonstationary spatial data using piecewise Gaussian processes. *Journal of the American Statistical Association*, 100(470), 653–668.

Krause, A., Singh, A. and Guestrin, C. (2008) Near-optimal sensor placements in Gaussian processes: Theory, efficient algorithms and empirical studies. *Journal of Machine Learning Research*, 9(1), 235–284.

Mack, Y. and Rosenblatt, M. (1979) Multivariate k-nearest neighbor density estimates. *Journal of Multivariate Analysis*, 9(1), 1–15.

Malloy, M.L. and Nowak, R.D. (2014) Near-optimal adaptive compressed sensing. *IEEE Transactions on Information Theory*, 60(7), 4001–4012.

Nikolaev, P., Hooper, D., Perea-Lopez, N., Terrones, M. and Maruyama, B. (2014) Discovery of wall-selective carbon nanotube growth conditions via automated experimentation. *ACS Nano*, 8(10), 10214–10222.

Nikolaev, P., Hooper, D., Webber, F., Rao, R., Decker, K., Krein, M., Poleski, J., Barto, R. and Maruyama, B. (2016) Autonomy in materials research: A case study in carbon nanotube growth. *npj Computational Materials*, 2(16031), 1–6.

Paisley, J., Liao, X. and Carin, L. (2010) Active learning and basis selection for kernel-based linear models: A Bayesian perspective. *IEEE Transactions on Signal Processing*, 58(5), 2686–2700.

Park, C. and Ding, Y. (2019) Automating material image analysis for material discovery. *MRS Communications*, 9(2), 545–555.

Park, C. and Ding, Y. (2021) *Data Science for Nano Image Analysis*. Springer, Switzerland.

Pope, C.A., Gosling, J.P., Barber, S., Johnson, J.S., Yamaguchi, T., Feingold, G. and Blackwell, P.G. (2021) Gaussian process modeling of heterogeneity and discontinuities using Voronoi tessellations. *Technometrics*, 63(1), 53–63.

Qiu, P. (1998) Discontinuous regression surfaces fitting. *The Annals of Statistics*, 26(6), 2218–2245.

Qiu, P. (2004) The local piecewisely linear kernel smoothing procedure for fitting jump regression surfaces. *Technometrics*, 46(1), 87–98.

Qiu, P. (2005) *Image Processing and Jump Regression Analysis*. John Wiley & Sons, Hoboken, NJ.

Qiu, P. (2009) Jump-preserving surface reconstruction from noisy data. *Annals of the Institute of Statistical Mathematics*, 61(3), 715–751.

Qiu, P. and Yandell, B. (1997) Jump detection in regression surfaces. *Journal of Computational and Graphical Statistics*, 6(3), 332–354.

Ranj, P., Bingham, D. and Michaillidis, G. (2008) Sequential experiment design for contour estimation from complex computer codes. *Technometrics*, 50(4), 527–541.

Rao, R., Liptak, D., Cherukuri, T., Yakobson, B.I. and Maruyama, B. (2012) In situ evidence for chirality-dependent growth rates of individual carbon nanotubes. *Nature Materials*, 11(3), 213–216.

Sacks, J., Welch, W.J., Mitchell, T.J. and Wynn, H.P. (1989) Design and analysis of computer experiments. *Statistical Science*, 4(4), 409–423.

Singh, A., Krause, A., Guestrin, C. and Kaiser, W.J. (2009) Efficient informative sensing using multiple robots. *Journal of Artificial Intelligence Research*, 34(1), 707–755.

Stach, E., DeCost, B., Kusne, A.G., Hattrick-Simpers, J., Brown, K.A., Reyes, K.G., Schrier, J., Billing, S., Buonassisi, T., Foster, I., Gomes, C.P., Gregoire, J.M., Mehta, A., Montoya, J., Olivetti, E., Park, C., Rotenberg, E., Saikin, S.K., Smullin, S., Stanef, V. and Maruyama, B. (2021) Autonomous experimentation systems for materials development: A community perspective. *Nature*, 590(7847), 2702–2726.

Stevens, A., Kovarik, L., Abellan, P., Yuan, X., Carin, L. and Browning, N.D. (2015) Applying compressive sensing to TEM video: A substantial frame rate increase on any camera. *Advanced Structural and Chemical Imaging*, 1(10), 1–20.

Taddy, M.A., Gramacy, R.B. and Polson, N.G. (2011) Dynamic trees for learning and design. *Journal of the American Statistical Association*, 106(493), 109–123.

Wasserman, L. (2006) *All of Nonparametric Statistics*. Springer-Verlag, New York, NY.

Zhao, Z. and Yao, W. (2012) Sequential design for nonparametric inference. *Canadian Journal of Statistics*, 40(2), 362–377.

Zhu, Z. and Stein, M.L. (2006) Spatial sampling design for prediction with estimated parameters. *Journal of Agricultural, Biological, and Environmental Statistics*, 11(1), 24–44.

Zimmerman, D.L. (2006) Optimal network design for spatial prediction, covariance parameter estimation, and empirical prediction. *Environmetrics*, 17(6), 635–652.