Journal of Quality Technology

A Quarterly Journal of Methods, Applications and Related Topics

ISSN: (Print) (Online) Journal homepage: https://www.tandfonline.com/loi/ujqt20

Computationally efficient Bayesian sequential function monitoring

Wright Shamp , Roumen Varbanov , Eric Chicken , Antonio Linero & Yun Yang

JOURNAL OF QUALITY TECHNOLOGY
https://doi.org/10.1080/00224065.2020.1801366

Computationally efficient Bayesian sequential function monitoring

Wright Shamp[a], Roumen Varbanov[a], Eric Chicken[a], Antonio Linero[a,b], and Yun Yang[a,c]

[a]Statistics, Florida State University, Tallahassee, Florida; [b]Department of Statistics and Data Sciences, University of Texas at Austin, Austin, Texas; [c]Department of Statistics, University of Illinois at Urbana-Champaign, Urbana, Illinois

ABSTRACT

In functional sequential process monitoring, a process is characterized by sequences of observations called profiles which are monitored over time for stability. The goal is to halt a process when the process generating these observations deviates from a specified in control standard. We propose a Bayesian sequential process control (SPC) methodology which uses wavelets to monitor the functional responses and detect out of control profiles. Our contribution is to propose a solution to the growing computational cost by constructing an efficient and accurate approximation to the posterior distribution of the wavelet coefficients, without recourse to Markov chain Monte Carlo.

KEYWORDS

Bayesian; Phase II; profile monitoring; statistical process control; wavelets

1. Introduction

Statistical process control (SPC) techniques are commonly used to ensure the quality of a process over time. For example, in the context of manufacturing, SPC may be used to determine if and when a manufacturing process deviates from a desired quality level. We consider an observed sequence of discretely sampled functional profiles

$$y_i^t = f^t(x_i) + \epsilon_i^t, \quad i = 1, \ldots, n$$

where $\boldsymbol{y}^t = (y_1^t, \ldots, y_n^t)$ is a noise-contaminated quality characteristic profile, x_i is a covariate, n is the number of observations in a functional profile, and ϵ_i^t represents the noise associated with the process. For example, observations y_i^t can be measurements of a manufactured part at the corresponding positions x_i.

Examples of profile monitoring applications in the literature include roundness evaluation of mechanical components, where radial measurement is a function of turning (Colosimo, Semeraro, and Pacella 2008), and automobile engine testing, where torque produced is a function of engine speed (Amiri, Jensen, and Kazemzadeh 2009). SPC methods are carried out in two phases; this paper will focus on Phase II profile monitoring, which falls into an area of statistics known as quickest change detection (QCD).

We propose a wavelet-based Bayesian framework for Phase II non-linear profile monitoring. To avoid making assumptions about the functional form $f^t(x_i)$, we will use the wavelet-based approach to model the profiles introduced in Varbanov et al. (2019), where the full posterior probability of a change point occurring at each observed time point is computed.

A key limitation of performing exact change-point detection is that, given T observed profiles, computing the posterior probability that a change has occurred ostensibly requires $O(T)$ computations. Accordingly, as time proceeds, the overall computational requirement is on the order $O(\sum_{t=1}^{T} t) = O(T^2)$. This is a substantial drawback when it is desired to perform SPC with data arriving in real-time. Our primary contribution is to propose a solution to the problem of growing computational cost by constructing an efficient and accurate approximation of the posterior distribution of wavelet coefficients such that the total computational cost at time t is bounded. Our approximate posterior is constructed by "merging" together the posterior distribution at times which carry similar information about the underlying process. Our approach can be compared with other approximation methods, such as windowing. Windowing methods, such as those described by Willsky and Jones (1976) and Hawkins and Zamba (2005), assume at time T that a change has occurred only within the last W time-points. If the signal is sufficiently small, however, it may be the case that the window size is too small

CONTACT Wright Shamp ✉ wright.shamp@stat.fsu.edu Statistics, Florida State University, 214 Rogers Building (OSB), 117 N. Woodward Ave., P.O. Box 3064330, Tallahassee, FL 32306-4330, USA.

Supplemental data for this article is available online at https://doi.org/10.1080/00224065.2020.1801366

to detect a change with non-negligible probability. Relative to these methods, our approach has the benefit of approximately maintaining the entire history of the process, and allowing for the possibility of the change-point occurring far in the past.

The paper is organized as follows. In Section 2, we review relevant material on SPC methods, wavelet methods, and the wavelet-based Bayesian approach developed by Varbanov et al. (2019) as well as its computational limitations. In Section 3 we introduce our posterior approximation method and the merged Wavelet-based Bayesian (MWBB) method for sequential change-point detection. In Section 4 we compare the proposed method to existing methods through simulation, and apply the methodology to the vertical density profiling dataset of Winistorfer, Young, and Walker (1996). Section 5 concludes with a discussion of results and considerations for future work.

2. Review of relevant material

2.1. Statistical process control background and quickest change detection

SPC is traditionally done in two separate steps. The first step, Phase I, uses historical data to identify and summarize in control performance. The next step, Phase II, collects and analyzes new data to detect significant deviations from the in control performance established in Phase I. Statistically, Phase I focuses on parameter estimation based on data from the in control distribution, while Phase II focuses on performing inference using online methods that operate until the process is stopped. We focus primarily on Phase II of SPC, and assume that the in control behavior is known *a priori* (in-control behavior can also be estimated). This reduces the Phase II problem to performing the quickest change detection (QCD) problem in an online fashion.

Throughout this paper, we will consider the following model for the observed functional observations:

$$y_i^t = f^0(x_i) + g(x_i)I(t \geq \tau) + \epsilon_i^t, \quad \epsilon_i^t \overset{\text{iid}}{\sim} \mathcal{N}(0, \sigma_\epsilon^2) \tag{1}$$

where f^0 is the functional relationship between the explanatory variable and the quality characteristic when the process is in control and g is the unknown functional change introduced when the process goes out of control. The goal is to determine whether the process is in control given the profiles observed at times $t = 1, 2, \ldots, T$ and continuously classify profiles control state with each newly observed $\boldsymbol{y}^t = (y_1^t, \ldots, y_n^t)$. Once the process is stopped, we may also be interested in estimating τ, the time of the first out of control profile. In Phase II, f^0 and σ_ϵ^2 are assumed to either be known or are approximated, and hence can be taken without loss of generality to be $f^0(x_i) = 0$ and $\sigma_\epsilon^2 = 1$, respectively.

Performance of an SPC procedure can be measured by a tradeoff of allowing an in control process to continue to run and stopping an out of control process quickly. A false alarm occurs when a monitoring procedure flags an in control process as out of control. Monitoring procedures are typically assessed according to their ability to detect changes quickly, subject to maintaining a desired false alarm rate. Let $\hat{\tau}$ be a *stopping rule* such that the process is halted at time $\hat{\tau}$. Given the model (1), let $E_t(\cdot)$ denote the expectation operator for the model with $[\tau = t]$. We define the in control average run length (ARL_0) to be $E_\infty(\hat{\tau})$, which is the average time until a false alarm occurs when the process is in control. Subject to the constraint $E_\infty(\hat{\tau}) = \text{ARL}_0$ is a specific value, we then aim to find procedures with a small *detection delay* $E_\tau(\hat{\tau} - \tau | \hat{\tau} \geq \tau)$.

Bayesian approaches to the generic QCD problem have a long history, being initially developed by Shiryaev (1963). This perspective treats the τ as a random variable with a prior distribution $\pi(\tau)$. Bayesian procedures have been of general interest as a tool for deriving traditional (minimax-optimal) procedures. Assuming a zero-inflated geometric prior on τ, Shiryaev (1963) showed that the rule which stops when $\pi(\tau \leq T|$ observed data up to $T)$ exceeds a specified upper control limit U is optimal from a Bayesian perspective. Tartakovsky and Veeravalli (2005) extended this result, demonstrating that this rule is asymptotically optimal for a very broad class of changepoint models. For practical purposes, an online implementation of the Bayesian sequential change-point detection was developed by Adams and MacKay (2007). When the data generating mechanism lies in an exponential family, QCD can be performed efficiently by storing a collection of sufficient statistics.

2.2. Wavelet background

Wavelets are localized wave-like functions that can be translated and dilated to create a basis for a wide range of functions. Wavelets possess inherent spatial adaptivity, which is useful when analyzing functions with discontinuities and sharp spikes, as well as time-frequency localization. Wavelets are attractive as tools for function estimation because most functions *f*(*x*) can be well-approximated using a wavelet basis expansion in which most coefficients are zero. For a more complete introduction and overview of wavelets see Ogden (2012) and Vidakovic (2009).

Let ϕ and ψ denote the compactly-supported father and mother wavelet functions, respectively. The translations and dilations of ϕ and ψ given by

$$\phi_{jk}(x) = 2^{j/2}\phi(2^j x - k), \quad \psi_{jk}(x) = 2^{j/2}\psi(2^j x - k)$$

generate (for any fixed integer j_0) an orthonormal basis $\{\phi_{j_0k}, \psi_{jk} | j_0, j, k \in \mathbb{Z}; j \geq j_0\}$ for the space of square-integrable functions $L_2(\mathbb{R})$. A function $f \in L_2(\mathbb{R})$ can then be expressed as the infinite series

$$f(x) = \sum_k \xi_{j_0k}\phi_{j_0k}(x) + \sum_{j=j_0}^{\infty}\sum_k \theta_{jk}\psi_{jk}(x), \tag{2}$$

with wavelet coefficients $\xi_{j_0k} = \langle f, \phi_{j_0k}\rangle$ and $\theta_{jk} = \langle f, \psi_{jk}\rangle$ where $\langle f, g\rangle = \int f \ gdx$ is the usual inner product on $L_2(\mathbb{R})$. The first series in Eq. [2], consisting of translations of ϕ_{j_0k}, represent the smoothest structure of the function, while the second series in Eq. [2], consisting of translations and dilations of ψ_{jk}, represent the higher frequency parts of the function. We will refer to ξ_{j_0k} and θ_{jk} as smooth (also called coarse) and detail wavelet coefficients, respectively. Projecting a function onto the wavelet basis allows for a multiresolution analysis (Mallat 1989b). The variable j gives the resolution level, while the variable k represents the location along the x-axis.

In practice, functional data is observed discretely. Suppose we have $\boldsymbol{y} = \boldsymbol{f} + \boldsymbol{\epsilon}$ where $\boldsymbol{f} = (f(x_1), \ldots, f(x_n))^\top$ and $\boldsymbol{\epsilon} \sim \mathcal{N}(0, \sigma^2 I)$, such that the x_i's are evenly spaced and $n = 2^J$ is a dyadic integer. Then the pyramid algorithm described in Mallat (1989a) can be used to perform the Discrete Wavelet Transform (DWT) on $\boldsymbol{y}$ with $O(n)$ computations. This gives a vector of n estimated wavelet coefficients

$$\boldsymbol{d} = \boldsymbol{W}\boldsymbol{y} = (c_{j_0,0}, c_{j_0,1}, \ldots, c_{j_0,2^{j_0}-1}, d_{j_0,0}, d_{j_0,1}, \ldots, d_{J-1,2^{J-1}}).$$

The coefficients $c_{j_0,k} \approx \xi_{j_0,k}$ and $d_{j,k} \approx \theta_{j,k}$ are approximations of the coefficients from the wavelet series expansion in Eq. [2]. Due to the orthonormality of the basis, the linear transformation $\boldsymbol{W}$ encoding the DWT is orthogonal, so $\boldsymbol{d} \sim \mathcal{N}(\boldsymbol{\theta}, \sigma^2 I)$ where $\boldsymbol{\theta} = E(\boldsymbol{d})$ consists of the coefficients $\xi_{j_0,k}$ and θ_{jk}. For the remainder of the paper, the term wavelet coefficient will refer to the wavelet coefficient obtained by the DWT.

2.3. Wavelet-based Bayesian SPC

The discrete version of the SPC problem sets $\boldsymbol{y}^t = I(t \geq \tau)\boldsymbol{g} + \boldsymbol{\epsilon}^t$ where $\boldsymbol{\epsilon}^t \sim \mathcal{N}(0, I)$ and $\boldsymbol{g} = (g(x_1), \ldots, g(x_n))^\top$; recall that we have set $f^0(x_i) = 0$ and $\sigma_\epsilon^2 = 1$ without loss of generality and that τ represents the first time at which the process is out of control. Let $\boldsymbol{d}^t = \boldsymbol{W}\boldsymbol{y}^t$ denote the vector of wavelet coefficients obtained from the t^{th} observed profile. When the process is in control we have $\boldsymbol{d}^t = \boldsymbol{W}\boldsymbol{\epsilon}^t \sim \mathcal{N}(0, I)$; hence, the in control wavelet coefficients will contain only white noise. Once the process is out of control, we have $\boldsymbol{d}^t \sim \mathcal{N}(\boldsymbol{\theta}, I)$, where $\boldsymbol{\theta} = \boldsymbol{W}\boldsymbol{g}$ is the vector of true wavelet coefficients for the functional change $\boldsymbol{g}$. Varbanov et al. (2019) use priors on the wavelet coefficients $\boldsymbol{\theta}$ to flexibly estimate $\boldsymbol{g}$. Let $D^T = (\boldsymbol{d}^1, \ldots, \boldsymbol{d}^T)$ denote the empirical wavelet coefficients obtained up-to-and-including time T. We will use π to refer to the prior and posterior mass functions for $(\boldsymbol{\theta}, \tau)$ and we will use f to refer to marginal/conditional densities of the data. When a mass function for τ needs to be evaluated at a particular time t we will write $\pi(\tau = t)$. The marginal posterior distribution of the change point τ can be written

$$\pi(\tau | D^T) = \frac{\int f(D^T | \tau, \boldsymbol{\theta}) \ \pi(\tau) \ \pi(\boldsymbol{\theta}) d\boldsymbol{\theta}}{\sum_\tau \int f(D^T | \tau, \boldsymbol{\theta}) \ \pi(\tau) \ \pi(\boldsymbol{\theta}) d\boldsymbol{\theta}} \tag{3}$$

$$= \frac{f(D^T | \tau)\pi(\tau)}{\sum_{t=1}^T f(D^T | \tau = t)\pi(\tau = t) + f(D^T | \tau > T)\pi(\tau > T)}. \tag{4}$$

Varbanov et al. (2019) uses the *Shiryaev procedure* (Shiryaev 1963), which classifies the process as out of control if the monitoring statistic $\pi(\tau \leq T | D^T) = \sum_{t \leq T} \pi(\tau = t | D^T)$ exceeds a specified upper control limit (UCL) U. Varbanov et al. (2019) use a geometric prior $\pi(\tau) = (1 - p)^{\tau - 1} \ p$ with the mean $1/p$ giving the prior average time at which the process goes out of control.

To model the sparsity of the detail coefficients, they use a spike-and-slab prior (Abramovich, Sapatinas, and Silverman 1998; Johnstone and Silverman 2005). Sparsity in the wavelet representation is encoded through the mixture prior

$$\pi(\theta_{jk}) = (1 - \omega) \ \delta_0(\theta_{jk}) + \omega \ \mathcal{N}(\theta_{jk} | 0, s^2), \tag{5}$$

where δ_0 is the Dirac mass at zero and $\mathcal{N}(y | \mu, \sigma^2)$ is a normal density function centered at μ with scale σ.

Draws from the mixture prior (5) are zero with probability $1 - \omega$, and are normal with scale s otherwise. The normal density is used because it is a conjugate prior for the normal likelihood associated with the observed coefficients. For the coarse coefficients, we remove the point mass at zero, giving $\pi(\xi_{j0k}) = N(\xi_{j0k} | 0, s^2)$.

In order to compute the posterior distribution of τ, we need to compute the density of the observed coefficients $f(D^T|\tau)$. Integrating out $\boldsymbol{\theta}$, Varbanov et al. (2019) shows that this is given by

$$f(D^T|\tau) = \left[\prod_{t=1}^{\tau-1}\prod_{i=1}^{n}\mathcal{N}(d_i^t|0,1)\right]\left[\prod_{i=1}^{n}u(\boldsymbol{d}_i^{t\geq\tau})\ v(\bar{d}_i^{t\geq\tau})\right],$$

with

$$u(\boldsymbol{d}_i^{t\geq\tau}) = \frac{1}{\sqrt{n_\tau(2\pi)^{n_\tau-1}}}\exp\left(-\frac{\sum_{t=\tau}^{T}(d_i^t-\bar{d}_i^{t\geq\tau})^2}{2}\right),$$
$$v(\bar{d}_i^{t\geq\tau}) = (1-\omega)\ \mathcal{N}_{n_\tau^{-1}}(\bar{d}_i^{t\geq\tau}|0,n_\tau^{-1}) + \omega\ \mathcal{N}(\bar{d}_i^{t\geq\tau}|0,s^2+n_\tau^{-1}).$$

Here, $\boldsymbol{d}_i^{t\geq\tau} = (d_i^\tau, \ldots, d_i^T), \bar{d}_i^{t\geq\tau}$ is the average of $\boldsymbol{d}_i^{t\geq\tau}$ and $n_\tau = T-\tau+1$ is the number of profiles which are out of control given τ. This results in a tractable posterior distribution of the change-point probability. For the remainder of the paper, this method will be called full wavelet-based Bayesian (FWBB) method.

2.4. Calibration and hyperparameters

The FWBB procedure requires specification of the hyperparameters (ω, s, p). Previous works have considered setting the parameters ω and s via empirical Bayes, optimizing the marginal likelihood $f(D^T)$ — which depends implicitly on (ω, s, p) — with respect to the hyperparameters. Unfortunately, optimization of the hyperparameters is too costly in an online setting. Instead, we consider a heuristic approach where ω and s are chosen to match a standard thresholding rule. Specifically, Johnstone and Silverman (2005) note that the posterior median estimate of $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_n)$ corresponds to the soft-thresholding

$$\text{median}(\theta_i|\boldsymbol{d}) = \text{sign}(d_i)\ \max(0, h(d_i, \omega, s)).$$

The function h is given by

$$h(d_i, \omega, s) = \frac{s^2}{s^2+1}|d_i| \\ - \frac{s}{\sqrt{1+s^2}}\Phi^{-1}\left(\frac{1+\min(\omega_i, 1)}{2}\right),$$

where ω_i denotes the posterior odds of a coefficient being zero and Φ denotes the distribution function of a standard normal variable. These equations establish a relationship between ω, s, and the posterior median threshold, which is defined as $\lambda_{\text{med}} = \inf\{d \geq 0 : h(d, \omega, s) > 0\}$. Given a user-specified ω, Varbanov et al. (2019) select s so that it matches the universal threshold $\lambda_{\text{univ}} = \sqrt{2\log n}$ introduced by Donoho and Johnstone (1994). User can also specify s, and select ω such that it matches the universal threshold. Donoho and Johnstone (1994) discuss the efficiency of this thresholding procedure to select wavelet coefficients pertaining to signal while ignoring wavelet coefficients related to noise with high probability.

3. Merging components for posterior approximation

A fundamental issue with process monitoring methods is the cost of computing the monitoring statistic. At time T, a change point statistic requires $O(T)$ computations, and computing the statistic at times $t = 1, 2, \ldots, T$ requires $O(T^2)$ computations. This creates problems in functional profile monitoring, where the cost of computing the monitoring statistic is also a function of the dimension of the observation, which can be large. For the FWBB procedure discussed in Section 2.3, each possible change point requires $O(n)$ computations and the statistic $\pi(\tau \leq T|D^T)$ requires $O(Tn)$ computations for profiles of length n.

We solve the problem of growing computational cost by approximating the posterior distribution of $(\tau, \boldsymbol{\theta})$ at each time. The computational cost will be controlled by maintaining an approximation to $\pi(\boldsymbol{\theta}, \tau \in A|D^T)$ for A in a partition $\mathcal{A}$ of $\mathbb{N}$.

To motivate our procedure, note that for any partition $\mathcal{A}^{\text{Pres}}$ of $\{1, \ldots, T\}$ we can write $\pi(\tau \leq T|D^T) = \sum_{A\in\mathcal{A}^{\text{Pres}}}\pi(\tau \in A|D^T) = \sum_{A\in\mathcal{A}^{\text{Pres}}}\int\pi(\boldsymbol{\theta}, \tau \in A|D^T)d\boldsymbol{\theta}$; hence, computing $\pi(\boldsymbol{\theta}, \tau \in A|D^T)$ is sufficient to compute the monitoring statistic $\pi(\tau \leq T|D^T)$. We use the superscript Pres to denote the partition up to present time T. Additionally, we can compute the summation terms sequentially as

$$\pi(\tau \in A|D^T) \propto \pi(\tau \in A|D^{T-1})\ f(\boldsymbol{d}^T|\tau \in A, D^{T-1}).$$

This requires computing $|\mathcal{A}^{\text{Pres}}| + 1$ factors (one for each element of the summation, and one for $A = \{t : t > T\}$). By comparison, the non-partitioned expression $\pi(\tau \leq T|D^T) = \sum_{t=1}^{T}\pi(\tau = T|D^T)$ requires computing $T+1$ factors. Unfortunately, computing exactly $\pi(\tau \in A|D^{T-1})$ requires just as much work as computing the full posterior. For example,

$$f(\boldsymbol{d}^T|\tau \in A, D^{T-1}) = \sum_{t\in A}\pi(\tau = t|\tau \in A, D^{T-1})\ f(\boldsymbol{d}^T|\tau \\ = t, D^{T-1}),$$

so that we still need to compute $f(\boldsymbol{d}^T|\tau = t, D^{T-1})$ for each t. The distribution $f(\boldsymbol{d}^T|\tau \in A, D^{T-1})$ is a mixture of components that correspond to possible change-points for times $t \in A$. In the special case where $\pi(\boldsymbol{\theta}|\tau = t, D^{T-1})$ is identical for all $t \in A$ the mixture simplifies to $f(\boldsymbol{d}^T|\tau \in A, D^{T-1}) = f(\boldsymbol{d}^T|\tau = t, D^{T-1})$. This motivates iteratively building the partition in such a fashion that the distributions $f(\boldsymbol{d}^T|\tau = t, D^{T-1})$ are similar for all $t \in A$, and then approximating each term with the same *single* factor.

In principle, an ideal approximation would form the partition by grouping together times t for which $f(\boldsymbol{d}^T|\tau = t, D^{T-1})$ have minimal distance. We will see, however, that we can control the approximation error for $\pi(\tau \le T|D^T)$ by combining groups of times A and B for which $\pi(\tau \in A \cup B|D^T)$ is smaller than any other group of times (i.e. $\pi(\tau \in A \cup B|D^T) \le \pi(\tau \in C|D^T) \quad \forall C \in \mathcal{A}$). This is extremely convenient because $\pi(\tau \in A \cup B|D^T)$ is computed as a byproduct of computing $\pi(\tau \le T|D^T)$.

3.1. Algorithm

The algorithm we propose is given in Algorithm 1, with the specifics of each step laid out in the following sections.

Algorithm 1 Merged Wavelet-Based Bayesian QCD

```
Input: $T, \boldsymbol{d}^T, \mathcal{A}_T, \{p^{T-1}_{i,A}, \omega^{T-1}_{i,A}, m^{T-1}_{i,A}, \nu^{T-1}_{i,A} : A \in \mathcal{A}_T\}, U$

1. Compute $\tilde{f}(\boldsymbol{d}^T|t \in A)$ for $A \in \mathcal{A}_T$ according to (7).
2. Compute $p^T_A = \tilde{\pi}_{T-1}(\tau \in A|\boldsymbol{d}^T)$ for $A \in \mathcal{A}_T$ according to (8)
3. If $\sum_{A \in \mathcal{A}^{\text{Pres}}_T} p^T_A \ge U$ : Return out of control.
4. Choose $B \ne C \in \mathcal{A}^{\text{Pres}}_T$ so that $\max\{p^T_B, p^T_C\}$ is minimized.
5. Set $\mathcal{A}_{T+1}$ according to (10).
6. For $A \in \mathcal{A}^{\text{Pres}}_T$, compute $\omega^T_{i,A}, m^T_{i,A}, \nu^T_{i,A}$ according to (9) and set $\omega_{i,A} = \omega, m^T_{i,A} = 0$ and $\nu_{i,A} = 1$ for $A = \{T+1\}$ and $\{t : t > T\}$.
7. Compute $\omega^T_{i,B\cup C}$, $m^T_{i,B\cup C}$, and $\nu_{i,B\cup C}$ as in (11) and (12).
8. Return $\mathcal{A}_{T+1}$, $\{\omega^T_{i,A}, m^T_{i,A}, \nu^T_{i,A} : A \in \mathcal{A}_{T+1}\}$.
```

3.1.1. Notation

At time T, we maintain an approximation to the distribution $\pi(\boldsymbol{\theta}, \tau \in A|D^{T-1})$ denoted by $\tilde{\pi}_{T-1}(\boldsymbol{\theta}, \tau \in A)$. This approximation is stored only for $A \in \mathcal{A}_T$ where $\mathcal{A}_T$ is a partition of $\mathbb{N} = \{1, 2, \ldots\}$. We assume $\{T\}, \{t : t > T\} \in \mathcal{A}_T$. Let $\mathcal{A}^{\text{Pres}}_T = \{A \in \mathcal{A}_T : A \ne \{t : t > T\}\}$ consist of the elements of $\mathcal{A}_T$ corresponding to the past and present times. Using similar rational as before, we take the approximation to have spike-and-slab form $\tilde{\pi}_{T-1}(\tau \in A) = p^{T-1}_A$ and

$$\tilde{\pi}_{T-1}(\boldsymbol{\theta}|\tau \in A) = \prod_{i=1}^{n} \Big\{(1 - \omega^{T-1}_{i,A})\delta_0(\theta_i) + \omega^{T-1}_{i,A}\mathcal{N}(\theta_i|m^{T-1}_{i,A}, \nu^{T-1}_{i,A})\Big\}. \tag{6}$$

As $\boldsymbol{d}^T$ arrives, we can compute an approximation to its predictive distribution $\tau \in A$ as

$$\begin{aligned}\tilde{f}(\boldsymbol{d}^T|\tau \in A) &= \int \tilde{\pi}_{T-1}(\boldsymbol{\theta}|\tau \in A)\, f(\boldsymbol{d}^T|\tau \in A, \boldsymbol{\theta})d\boldsymbol{\theta}\\ &= \prod_{i=1}^{n}(1 - w^{T-1}_{i,A})\mathcal{N}(d^T_i|0,1)\\ &\quad + w^{T-1}_{i,A}\mathcal{N}(d^T_i|m^{T-1}_{i,A}, \nu^{T-1}_{i,A} + 1).\end{aligned} \tag{7}$$

This holds for $A \in \mathcal{A}^{\text{Pres}}_T$. For $A = \{T+1, T+2, \ldots\}$, the changepoint has not occurred, so we take $\tilde{f}(\boldsymbol{d}^T|\tau \in A) = \prod_{i=1}^{n}\mathcal{N}(d^T_i|0,1)$.

3.2. Updating the monitoring statistic

For the moment, we assume that we are at time T and have already in-hand the approximation $\tilde{\pi}_{T-1}$. In order to flag the process as out of control at time T we need to compute the statistic $\pi(\tau \le T|D^T)$. In terms of sequential updates, this can be written as

$$\begin{aligned}\pi(\tau \le T|D^T) &= \frac{\sum_{A \in \mathcal{A}^{\text{Pres}}_T} \pi(\tau \in A|D^{T-1})\, f(\boldsymbol{d}^T|\tau \in A, D^{T-1})}{\sum_{A \in \mathcal{A}_T} \pi(\tau \in A|D^{T-1})\, f(\boldsymbol{d}^T|\tau \in A, D^{T-1})}\\ &\approx \frac{\sum_{A \in \mathcal{A}^{\text{Pres}}_T} p^{T-1}_A\, \tilde{f}(\boldsymbol{d}^T|\tau \in A)}{\sum_{A \in \mathcal{A}_T} p^{T-1}_A\, \tilde{f}(\boldsymbol{d}^T|\tau \in A)}.\end{aligned}$$

Hence we can approximate the monitoring statistic given $\boldsymbol{d}^T$ by computing (7) and using the above formula.

3.3. Sequential updating

We now describe how to compute an approximation $\tilde{\pi}_T(\boldsymbol{\theta}, \tau \in A)$ for $A \in \mathcal{A}_T$. First, using the same approximation as for the monitoring statistic, we have

$$p^T_A = \frac{p^{T-1}_A\, \tilde{f}(\boldsymbol{d}^T|\tau \in A)}{\sum_{A \in \mathcal{A}_T} p^{T-1}_A\, \tilde{f}(\boldsymbol{d}^T|\tau \in A)}. \tag{8}$$

For $A = \{T+1, T+2, \ldots\}$, the changepoint has not occurred at time T. Conditional on the changepoint not occurring, the posterior $\pi(\boldsymbol{\theta}|\tau \in A, D^T)$ is exactly equal to the prior; hence, we set $m^T_{i,A} = 0, \nu^T_{i,A} = s^2$, and $\omega^T_{i,A} = \omega$. For $A \in \mathcal{A}^{\text{Pres}}_T$, we treat $\tilde{\pi}_{T-1}(\boldsymbol{\theta}, \tau \in A)$ as a prior at time T, giving the approximate model

$$\big[d^T_i|\boldsymbol{\theta}, \tau \in A\big] \overset{\text{ind}}{\sim} \mathcal{N}(\theta_i, 1), \qquad [\boldsymbol{\theta}|\tau \in A] \sim \tilde{\pi}_{T-1}(\boldsymbol{\theta}|\tau \in A), \qquad \tilde{\pi}_{T-1}(\tau \in A) = p^{T-1}_A.$$

We now apply Bayes rule to this approximate model. Using the conjugacy of the normal spike-and-slab prior, for $A \in \mathcal{A}^{\text{Pres}}_T$, we have

$$\tilde{\pi}_{T-1}(\boldsymbol{\theta}|\tau \in A, \boldsymbol{d}^T) = \prod_{i=1}^{n}(1 - \omega^T_{i,A})\delta_0(\theta_i) + \omega^T_{i,A}\mathcal{N}(\theta_i|m^T_{i,A}, \nu^T_{i,A})$$

where

$$m^T_{i,A} = \frac{m^{T-1}_{i,A} + \nu^{T-1}_{i,A} d^T_i}{\nu^{T-1}_{i,A} + 1}, \quad \nu^{T-1}_{i,A} = \frac{\nu^{T-1}_{i,A}}{\nu^{T-1}_{i,A} + 1},$$
$$\omega^T_{i,A} \propto \omega^{T-1}_{i,A} \mathcal{N}(d^T_i | m^{T-1}_{i,A}, \nu^{T-1}_{i,A} + 1). \tag{9}$$

3.4. Splitting

After $\tilde{\pi}_T(\boldsymbol{\theta}, \tau \in A)$ is constructed for these sets, we can now use splitting to construct $\mathcal{A}_{T+1}$ and the approximation for $A \in \mathcal{A}_{T+1}$. Let $B = \{T+1\}$ and $C = \{T+2, T+3, \ldots\}$. From the previous step, we have the approximate posterior $\tilde{\pi}_T(\boldsymbol{\theta}, \tau \in B \cup C)$. Since $B, C \in \mathcal{A}_{T+1}$, we instead need $\tilde{\pi}_T(\boldsymbol{\theta}, \tau \in B)$ and $\tilde{\pi}_T(\boldsymbol{\theta}, \tau \in C)$. Because the changepoint has not occurred for $\tau \in B \cup C$, we can take $\tilde{\pi}_T(\boldsymbol{\theta} | \tau \in B)$ and $\tilde{\pi}_T(\boldsymbol{\theta} | \tau \in C)$ to be exactly equal to their posterior distribution, i.e.,

$$m_{i,B} = m_{i,C} = 0, \quad \nu_{i,B} = \nu_{i,C} = s^2,$$
$$\omega_{i,B} = \omega_{i,C} = \omega.$$

To approximate $\pi(\tau \in B | D^T)$, we note that

$$\pi(\tau \in B | D^T) = \pi(\tau \geq T+1 | D^T)$$
$$\times \pi(\tau \in B | \tau \geq T+1).$$

This holds because, given that the changepoint has not occurred, D^T carries no information about when the process will go out of control. Approximating $\pi(\tau \geq T | D^T) \approx p^T_{B\cup C}$ from the previous step, we take

$$p^T_B = p^T_{B\cup C} \times \frac{\pi(\tau = T+1)}{\pi(\tau \geq T+1)}, \quad \text{and} \quad p^T_C$$
$$= p^T_{B\cup C} \times \frac{1 - \pi(\tau = T+1)}{\pi(\tau \geq T+1)}.$$

When τ is geometric with success probability p, this simplifies to $p^T_B = p^T_{B\cup C} \times p$ and $p^T_C = p^T_{B\cup C} \times (1-p)$.

3.5. Merging

Now, fix $B, C \in \mathcal{A}^{\text{Pres}}_T$, which are selected according to some as-yet unspecified criteria. To maintain a fixed computational budget, we require $|\mathcal{A}_{T+1}| \leq K_{\max} + 2$ for some user-specified $K_{\max.}$ The choice of $K_{\max}$ can be based on balancing approximation accuracy and computational cost allowance. The effect of different choices of $K_{\max}$ is discussed in Section 4. If we formed $\mathcal{A}_{T+1}$ by only splitting $\{T+1, T+2, \ldots\}$ into $\{T+1\}$ and $\{T+2, T+3, \ldots\}$, we would increase $|\mathcal{A}_{T+1}|$ by 1 every iteration. To preserve a fixed size for $\mathcal{A}_{T+1}$, we “merge” B and C together. That is, we set

$$\mathcal{A}_{T+1} = \{\mathcal{A} \in \mathcal{A}^{\text{Pres}}_T : \mathcal{A} \neq \mathcal{B}, \mathcal{C}\} \cup \{B \cup C\}$$
$$\cup \{\{T+1\}\} \cup \{\{t : t > T+1\}\}, \tag{10}$$

In order to be consistent with the approximation $\tilde{\pi}_T$ already constructed, we set $p^T_{B\cup C} = p^T_B + p^T_C$; unlike before, however, we cannot set $\tilde{\pi}_T(\boldsymbol{\theta} | \tau \in B \cup C) = \tilde{\pi}_{T-1}(\boldsymbol{\theta} | \tau \in B \cup C, \boldsymbol{d}^T)$, because $\tilde{\pi}_{T-1}(\boldsymbol{\theta} | \tau \in B \cup C, \boldsymbol{d}^T)$ is not a product of independent spike-and-slab distributions. In particular, we have

$$\tilde{\pi}_{T-1}(\boldsymbol{\theta} | \tau \in B \cup C, \boldsymbol{d}^T) = \frac{p^T_B}{p^T_B + p^T_C} \prod_{i=1}^{n}$$
$$\left\{(1 - \omega^T_{i,B})\delta_0(\theta_i) + \omega^T_{i,B} \mathcal{N}(\theta_i | m^T_{i,B}, \nu^T_{i,B})\right\}$$
$$+ \frac{p^T_C}{p^T_B + p^T_C} \prod_{i=1}^{n} \left\{(1 - \omega^T_{i,C})\delta_0(\theta_i)\right.$$
$$\left. + \omega^T_{i,C} \mathcal{N}(\theta_i | m^T_{i,C}, \nu^T_{i,C})\right\}.$$

This is a mixture of two spike-and-slab distributions. Our goal is to approximate this mixture with a *single* spike-and-slab distribution. To do this, we consider the marginal posterior of θ_i:

$$\tilde{\pi}_{T-1}(\theta_i | \tau \in B \cup C, \boldsymbol{d}^T)$$
$$= \left(\frac{p^T_B(1 - \omega^T_{i,B})}{p^T_B + p^T_C} + \frac{p^T_C(1 - \omega^T_{i,C})}{p^T_B + p^T_C}\right)\delta_0(\theta_i)$$
$$+ \frac{p^T_B \omega^T_{i,B}}{p^T_B + p^T_C} \mathcal{N}(\theta_i | m^T_{i,B}, \nu^T_{i,B})$$
$$+ \frac{p^T_C \omega^T_{i,C}}{p^T_B + p^T_C} \mathcal{N}(\theta_i | m^T_{i,C}, \nu^T_{i,C}).$$

This is a mixture of three distributions: a point mass at 0, and two normal distributions. We match the weight assigned to 0 by setting

$$\omega^T_{i,B\cup C} = \frac{p^T_B \omega^T_{i,B}}{p^T_B + p^T_C} + \frac{p^T_C \omega^T_{i,C}}{p^T_B + p^T_C}. \tag{11}$$

Given that $\theta_i \neq 0$, θ_i has a marginal posterior which is a normal mixture

$$\frac{p^T_B \omega^T_{i,B}}{p^T_B \omega^T_{i,B} + p^T_C \omega^T_{i,C}} \mathcal{N}(\theta_i | m^T_{i,B}, \nu^T_{i,B})$$
$$+ \frac{p^T_C \omega^T_{i,C}}{p^T_B \omega^T_{i,B} + p^T_C \omega^T_{i,C}} \mathcal{N}(\theta_i | m^T_{i,C}, \nu^T_{i,C}).$$

We approximate this mixture with a single normal distribution by matching the first two moments of the mixture to the moments of the approximating normal.

This gives

$$m^T_{i,B\cup C} = \frac{p^T_B\omega^T_{i,B}}{p^T_B\omega^T_{i,B}+p^T_C\omega^T_{i,C}}m^T_{i,B} + \frac{p^T_C\omega^T_{i,C}}{p^T_B\omega^T_{i,B}+p^T_C\omega^T_{i,C}}m^T_{i,C}\nu^T_{i,B\cup C} = \frac{p^T_B\omega^T_{i,B}}{p^T_B\omega^T_{i,B}+p^T_C\omega^T_{i,C}}\nu^T_{i,B} + \frac{p^T_C\omega^T_{i,C}}{p^T_B\omega^T_{i,B}+p^T_C\omega^T_{i,C}}\nu^T_{i,C} + \frac{p^T_B\omega^T_{i,B}p^T_C\omega^T_{i,C}}{(p^T_B\omega^T_{i,B}+p^T_C\omega^T_{i,C})^2}(m^T_{i,B}-m^T_{i,C})^2. \tag{12}$$

This approximation is carried out whenever $T > K_{\max}$. For $T \le K_{\max}$, this approximation is not necessary, as we can simply take $\mathcal{A}_{T+1} = \{\{1\},\{2\},\dots,\{T+1\},\{t : t \ge T+2\}\}$ and maintain the posterior distribution exactly.

3.6. Justification of merging approximation

The selection of (11) and (12) can be justified as minimizing the Kullback-Leibler divergence from $\tilde{\pi}_{T-1}(\boldsymbol{\theta}|\tau \in B\cup C, \boldsymbol{d}^T)$ to $\tilde{\pi}_T(\boldsymbol{\theta}|\tau \in B\cup C)$. We consider the problem of finding an optimal approximation to $\tilde{\pi}_{T-1}(\boldsymbol{\theta}|\tau \in B\cup C, \boldsymbol{d}^T)$ within the family $\mathcal{Q} = \{q(\boldsymbol{\theta})\}$ subject to the condition that $q(\boldsymbol{\theta})$ is a spike-and-slab prior

$$q(\boldsymbol{\theta}) = \prod_{i=1}^n (1-\kappa_i)\delta_0(\theta_i) + \kappa_i\mathcal{N}(\theta_i|\xi_i,\rho_i). \tag{13}$$

A commonly used measure of how close a given distribution G is to another distribution H is the *Kullback-Leibler divergence* from G to H, given by $\mathrm{KL}(G||H) = \int \log\left(\frac{dG}{dH}\right)dG$ where dG/dH is the Radon-Nikodym derivative of G with respect to H. The following proposition, which is proved in the Supplementary Material, implies that our choice of $\tilde{\pi}_T(\boldsymbol{\theta}|\tau \in B\cup C)$ is optimal in the sense that it minimizes the Kullback-Leibler divergence from $\tilde{\pi}_{T-1}(\boldsymbol{\theta}|\tau \in B\cup C, \boldsymbol{d}^T)$ to $q(\boldsymbol{\theta})$.

Proposition 1. *Let $\pi(\boldsymbol{\theta})$ denote a mixture of spike-and-slab distributions*

$$\pi(\boldsymbol{\theta}) = \lambda\prod_{i=1}^n\{(1-\omega_{i1})\delta_0(\theta_i) + \omega_{i1}\mathcal{N}(\theta_i|m_{i1},\nu_{i1})\} + (1-\lambda)\prod_{i=1}^n\{(1-\omega_{i2})\delta_0(\theta_i) + \omega_{i2}\mathcal{N}(\theta_i|m_{i2},\nu_{i2})\},$$

and consider the family $\mathcal{Q}$ of densities of the form (13). Then the Kullback-Leibler divergence $\mathrm{KL}(\pi||q)$ *is minimized at*

$$\kappa_i = \lambda\omega_{i1} + (1-\lambda)\omega_{i2},$$
$$\xi_i = \frac{\lambda\omega_{i1}}{\lambda\omega_{i1}+(1-\lambda)\omega_{i2}}m_{i1} + \frac{(1-\lambda)\omega_{i2}}{\lambda\omega_{i1}+(1-\lambda)\omega_{i2}}m_{i2},$$
$$\rho_i = \frac{\lambda\omega_{i1}}{\lambda\omega_{i1}+(1-\lambda)\omega_{i2}}\nu_{i1} + \frac{(1-\lambda)\omega_{i2}}{\lambda\omega_{i1}+(1-\lambda)\omega_{i2}}\nu_{i2} + \frac{\lambda\omega_{i1}(1-\lambda)\omega_{i2}}{(\lambda\omega_{i1}+(1-\lambda)\omega_{i2})^2}(m_{i1}-m_{i2})^2.$$

Minimization of divergences such as $\mathrm{KL}(\pi||q)$, where π is a target distribution and q is an approximation, covers many approximate inference techniques used in practice. The approximating $q(\theta)$ chosen according to this criterion is required to cover the entire support of π which, when combined with the factorization assumption $q(\boldsymbol{\theta}) = \prod_i q_i(\theta_i)$, results in q being more diffuse than π. This can be contrasted with the commonly used variational Bayes approach (see Blei, Kucukelbir, and McAuliffe 2017 for a review), which considers the minimization of $\mathrm{KL}(q||\pi)$; this tends to result in approximations q which are more highly concentrated than π. Minimization of $\mathrm{KL}(\pi||q)$ has connections with expectation propagation algorithms (see, e.g., Bishop 2006, Chapter 10). We also note that our merging algorithm can be cast as a mixture reduction algorithm where the mixing weights are given by $\pi(\tau = t|D^T)$ and the mixture components are given by $\pi(\boldsymbol{\theta}|\tau = t, D^T)$. In that context, our approach is similar in spirit to the approach of Runnalls (2007), which merges mixture components in a Gaussian mixture model by minimizing $\mathrm{KL}(\pi||q)$.

3.7. Selection of components to merge

One way of understanding the approximation scheme above is that, at time T, we select $B, C \in \mathcal{A}^{\text{Pres}}_T$ and replace $\tilde{\pi}_{T-1}(\boldsymbol{\theta}|\tau \in B, \boldsymbol{d}^T)$ and $\tilde{\pi}_{T-1}(\boldsymbol{\theta}|\tau \in C, \boldsymbol{d}^T)$ with a single term $\tilde{\pi}_{T-1}(\boldsymbol{\theta}|\tau \in B\cup C, \boldsymbol{d}^T)$, which is then approximated using the spike-and-slab approximation $\tilde{\pi}_T(\boldsymbol{\theta}|\tau \in B\cup C)$. That is, the merging algorithm is equivalent to setting

$$\tilde{\pi}_T(\boldsymbol{\theta}|\tau \in B) \equiv \tilde{\pi}_T(\boldsymbol{\theta}|\tau \in C) \equiv \tilde{\pi}_T(\boldsymbol{\theta}|\tau \in B\cup C).$$

Ideally, we should aim to have $\tilde{\pi}_T(\boldsymbol{\theta}, \tau \in A)$ close to $\tilde{\pi}_{T-1}(\boldsymbol{\theta}, \tau \in A|\boldsymbol{d}^T)$. For example, we might aim to minimize the L_1 distance

$$\sum_A \int |\tilde{\pi}_T(\boldsymbol{\theta}, \tau \in A) - \tilde{\pi}_{T-1}(\boldsymbol{\theta}, \tau \in A|\boldsymbol{d}^T)|.$$

For all terms except for the B and C to be merged, the terms of the summation cancel exactly; hence, this reduces to

$$\begin{aligned}\Delta_T(B,C) = \tilde{\pi}_{T-1}(\tau \in B|\boldsymbol{d}^T) \int |\tilde{\pi}_T \\ (\boldsymbol{\theta}|\tau \in B \cup C) - \tilde{\pi}_{T-1}(\boldsymbol{\theta}|\tau \in B, \boldsymbol{d}^T)|d\boldsymbol{\theta} \\ + \tilde{\pi}_{T-1}(\tau \in C|\boldsymbol{d}^T) \int |\tilde{\pi}_T(\boldsymbol{\theta}|\tau \in B \cup C) \\ - \tilde{\pi}_{T-1}(\boldsymbol{\theta}|\tau \in C, \boldsymbol{d}^T)|d\boldsymbol{\theta}.\end{aligned}$$

In principle $\Delta_T(B,C)$ could be minimized directly over all possible $B, C \in \mathcal{A}_T^{\text{Pres}}$ to find the optimal sets to merge. We adopt a simpler approach by noting that, from the triangle inequality,

$$\Delta_T(B,C) \leq 4\max\{\tilde{\pi}_{T-1}(\tau \in B|\boldsymbol{d}^T), \tilde{\pi}_{T-1}(\tau \in C|\boldsymbol{d}^T)\}. \tag{14}$$

This holds independently of the quality of the approximation $\tilde{\pi}_T(\boldsymbol{\theta}|\tau \in B \cup C)$. Therefore, in practice, we take B and C to minimize the right hand side of (14); that is, we take B and C corresponding to the least-likely sets to contain τ. This approach has the advantage of being fast and automatic, while producing surprisingly accurate results in the simulation settings we examined.

As mentioned previously, the problem of selecting which B and C to merge is similar in spirit to the *mixture reduction* problem, which has been widely studied in the case of Gaussian mixtures; see Crouse et al. (2011) for a review. In principle, a more refined approach to selection of components to merge could be developed along similar lines using, for example, clustering algorithms to select times to merge (see Schieferdecker and Huber 2009).

$$\begin{aligned}\pi(\tau|D_W^T) &= \frac{f(D_W^T|\tau)\pi_\tau(\tau)}{\sum_{t=1}^{T} f(D_W^T|\tau = t)\pi_\tau(\tau = t) + f(D_W^T|\tau > T)\pi_\tau(\tau > T)} \\ &= \frac{f(D_W^T|\tau)\pi_\tau(\tau)}{f(D_W^T|\tau \leq T_1)\pi_\tau(\tau \leq T_1) + \sum_{t=T_2}^{T_W} f(D_W^T|\tau = t)\pi_\tau(\tau = t) + f(D_W^T|\tau > T)\pi_\tau(\tau > T)}.\end{aligned} \tag{15}$$

4. Results

We evaluate our approximation in three areas: (1) stability when monitoring an in control process, (2) how quickly we detect an out of control process, and (3) quality of the approximation to the ideal/exact monitoring statistic. We compare the performance of the merging wavelet Bayesian method to the exact full wavelet-based Bayesian method, as well as to the likelihood ratio method proposed by Chicken, Pignatiello, and Simpson (2009). Throughout, we refer the exact full wavelet-based Bayesian method as FWBB, and the approximate method based on merging as MWBB. We refer to the likelihood ratio based method as LRT. This comparison is of particular interest because both methods use wavelets to address the Phase II profile monitoring problem-one with a Bayesian approach, the other with a Frequentist approach. Both methods operate under the assumptions of i.i.d. Gaussian errors and require that the in control profile is known or estimated prior to conducting Phase II analysis. Also, both methods use the universal threshold for wavelet coefficients and a single sustained change in the structure of profiles. The parameters of simulation are similar to those used in Chicken, Pignatiello, and Simpson (2009), where the LRT outperformed similar methods described in Fan (1996), Jeong, Lu, and Wang (2006) and Jin and Shi (2001).

We also compare the performance of our proposed methodology to the classic profile monitoring approach of windowing. This windowing approach is discussed in Willsky and Jones (1976) and Hawkins and Zamba (2005). With this approach only the most recent data is considered when computing the posterior. Let W denote the window size in number of time steps and let $T_1, T_2, \ldots, T_W$ denote the respective time indices of profiles in the window at time T (note that $T_1 = T - W + 1$ and $T_W = T$). The windowed posterior, or the posterior given only the data $D_W^T := \{\boldsymbol{d}^t : t = T_1, T_2, \ldots, T_W\}$, is then

Since $f(D_W^T|\tau = t)$ is constant for $t \leq T_1$, we can quickly compute $f(D_W^T|\tau \leq T)$ by considering only $t = T_1$, eliminating most of the cost of computing the denominator. We will use window size of 10, based on a balance between computation speed and approximation accuracy. We refer to this method as Window 10.

Lastly, we will compare the performance of our proposed MWBB procedure to the monitoring

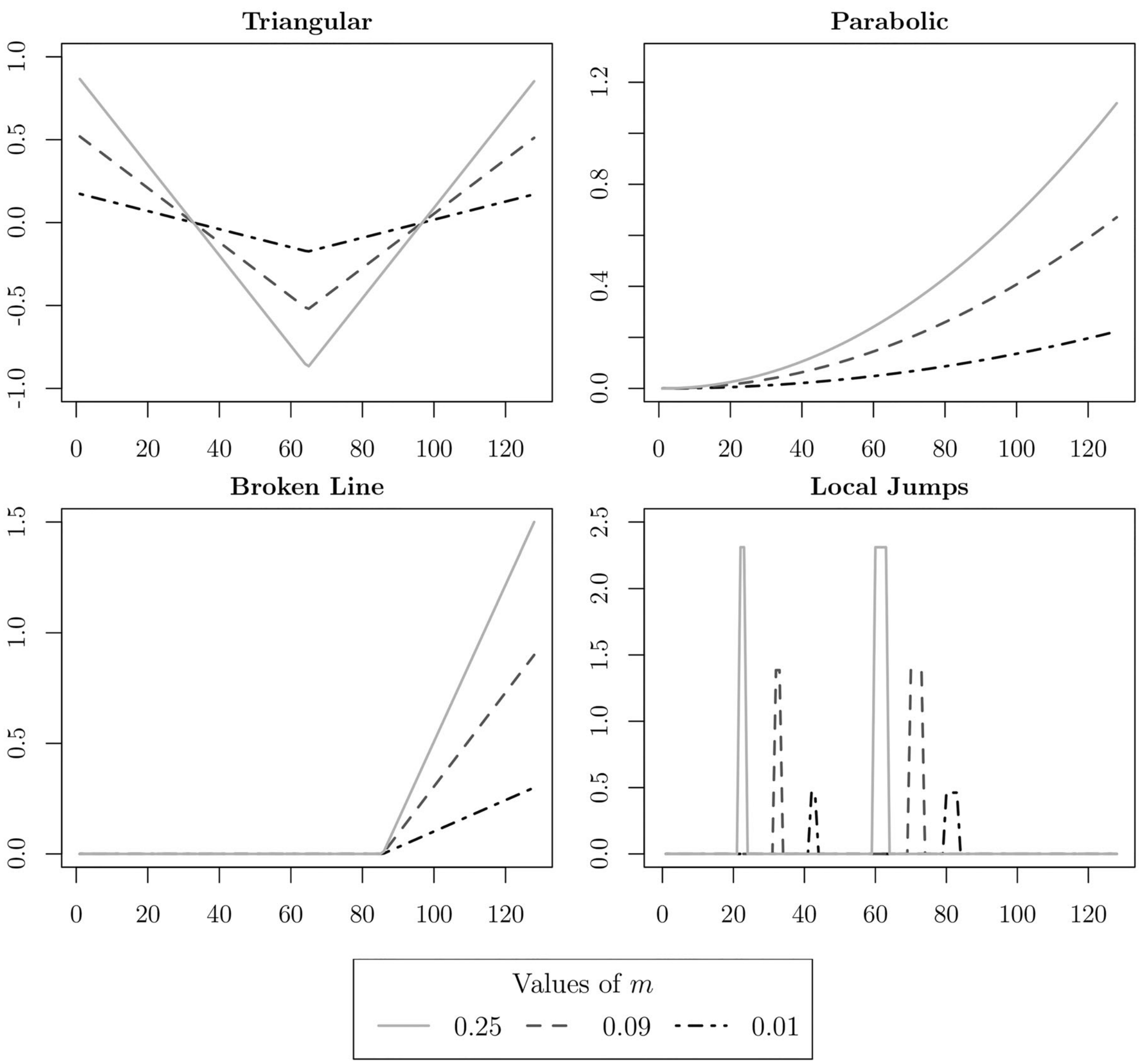

Figure 1. Different forms of functional change g with different magnitudes.

procedure presented in Jeong, Lu, and Wang (2006). This approach examines a subset of the wavelet coefficients using a data-adaptive thresholding procedure. We will refer to this methodology as the M^* method.

4.1. Simulations

We first evaluate our approximation under simulated settings. We show that the upper control limits to obtain a desired average in control run length are consistent across levels of $K_{\max}$ and that the monitoring statistic approximation is well-approximated when the process is in control. We then show that, under a variety of fault conditions, the MWBB procedure obtains similar performance to the FWBB procedure.

4.1.1. Test functions

An effective monitoring procedure should perform well under a variety of fault conditions. We consider several different functional changes g at different magnitudes m. The functional changes we consider are defined in Chicken, Pignatiello, and Simpson (2009) and are given in Figure 1; we refer to these as *triangular*, *parabolic*, *broken line*, and *local jumps* test functions. These test functions represent possible out of control scenarios. These test functions are scaled to give a target magnitude of change, which we define as

$$m = ||g||_2^2 = \int g^2 dx,$$

which is the squared L_2 norm of g.

4.1.2. Calibration

We first examine the MWBB procedure's sensitivity to the value of $K_{\max}$ during the calibration of the hyperparameters.

Table 1 gives calibration summaries for different values of $K_{\max}$ for the MWBB, FWBB, LRT, and

Table 1. A calibration summary of in control performance for different values of $K_{\max}$ based on 250 replications, $n = 128$ and $p = \frac{1}{100}$.

Method	ω	s	UCL	ARL$_0$	SDRL$_0$
$K_{max} = 5$	0.05	1.74	0.17	100.24	92.91
	0.10	1.07	0.25	100.46	98.58
	0.25	0.61	0.29	100.42	94.81
$K_{max} = 10$	0.05	1.74	0.18	100.01	92.80
	0.10	1.07	0.21	100.56	99.94
	0.25	0.61	0.29	100.56	94.18
$K_{max} = 20$	0.05	1.74	0.18	100.06	89.90
	0.10	1.07	0.21	100.16	97.62
	0.25	0.61	0.27	100.11	98.35
Full	0.05	1.74	0.17	100.07	88.85
	0.10	1.07	0.21	100.16	97.62
	0.25	0.61	0.27	100.24	95.86
LRT	0.05	1.74	0.07	100.61	98.66
	0.10	1.07	0.07	100.21	106.26
	0.25	0.61	0.07	100.61	98.67
Window 10	0.05	1.74	0.30	100.24	57.46
	0.10	1.07	0.34	100.32	66.73
	0.25	0.61	0.39	100.03	71.02

Windowed Wavelet procedures with $p = 1/100$ and ARL $= 100$. Results for $p = 1/370$ and 1/500 are given in the Supplementary Material.

Given this target ARL$_0$ we choose the UCL to be the smallest value for which the mean of the 250 simulated run lengths is at least the target ARL$_0$. We also report the standard deviation of the run length for each procedure, denoted as SDRL$_0$.

We see from Table 1 that the UCL is insensitive to the number of components carried forward (K_{max}) in the approximation procedure (small differences can be attributed to random error and are not statistically significant). The UCL for the MWBB method is sensitive to ω; this can attributed to more non-zero wavelet coefficients being monitored which introduces more noise into the profiles. This additional noise requires a higher UCL to obtain the desired ARL. From the tables in the Supplementary Material, we also see that the UCL is not affected by the choice of p for the FWBB and MWBB methods, although it does influence the windowing method.

4.1.3. Computation time

We now study the computational gains from using the merging procedure. Because FWBB requires $O(T^2)$ computations while MWBB requires $O(KT)$ computations, we expect that MWBB will be faster by a factor of $O(T/K)$. The Window Procedure is comparably fast, but we will see that it has lower power to detect out of control profiles. Results are given in Table 2 for $p = 1/100$. Results are similar for $p = 1/370$ and are given in the Supplementary Material.

Next we generate a sequence of 1000 in control profiles, with $K_{\max}$ varying from 1 to 500. Figure 2 shows the computation time associated with each procedure. We see that computation time is growing as $K_{\max}$ increases. The growth is not quite linear due to boundary effects as $K_{\max}$ approaches T; for example, if $K_{\max} = T$ then the merging algorithm will never activate and the computation will be the same as FWBB. This holds true for the four different parameter settings. Again, the windowing procedure is competitive in terms of computational efficiency; in order to justify the use of MWBB relative to windowing, we will analyze the approximation accuracy and detection delay of MWBB relative to windowing.

Table 2. A comparison of FWBB vs MWBB procedures computation time for $p = \frac{1}{100}$ with the length of profile sequence varying.

	Ratio of Comp Time for FWBB vs MWBB				
$K_{\max}$	$T = 100$	$T = 200$	$T = 300$	$T = 400$	$T = 500$
5	4.26	10.74	13.61	18.18	21.34
10	2.38	6.05	7.53	10.19	12.62
20	1.33	3.04	4.07	5.44	6.77
Window 10	4.56	8.55	13.80	17.87	22.48

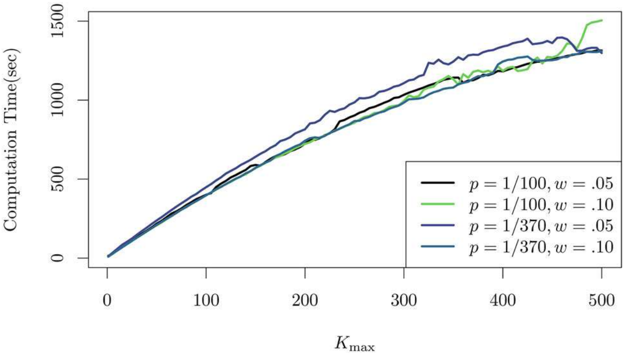

Figure 2. Computation time of MWBB procedure while varying K_{max}.

4.1.4. Approximation quality for in control settings

We now examine how well MWBB approximates the posterior obtained by FWBB, as measured by the L_1 difference between $\pi(\tau \le T|D^T)$ and $\tilde{\pi}_T(\tau \le T)$. We show two different parameter settings for p and use $(\omega, s) = (0.05, 1.74)$. We base the results on 250 replications of 500 profiles of length $n = 128$ generated under the in control setting. Results are given in Tables 3 and 4.

We see that the error is quite small for the MWBB procedures (less than 1% difference in probability of change point), even for $K_{\max}$ as small as 5, with the error decreasing as $K_{\max}$ increases. The windowing procedure has a much larger error, due to data being discounted outside of the defined window size. We also note that, as p decreases, the differences become substantially smaller, suggesting that the ideal

Table 3. A comparison of FWBB vs MWBB procedures performance for $p = \frac{1}{100}$ with the length of profile sequence varying.

	Mean Absolute Error				
$K_{\max}$	$T=100$	$T=200$	$T=300$	$T=400$	$T=500$
5	0.00077	0.00218	0.00347	0.00564	0.01018
10	0.00020	0.00010	0.00212	0.00431	0.00829
20	0.00003	0.00025	0.00088	0.00186	0.00442
Window 10	0.00761	0.02471	0.05361	0.09729	0.15576

Table 4. A comparison of FWBB vs MWBB procedures performance for $p = \frac{1}{370}$ with the length of profile sequence varying.

	Mean Absolute Error				
$K_{\max}$	$T=100$	$T=200$	$T=300$	$T=400$	$T=500$
5	0.00018	0.00043	0.00045	0.00065	0.00093
10	0.00006	0.00021	0.00019	0.00043	0.00069
20	0.00001	0.00004	0.00010	0.00026	0.00026
Window 10	0.00195	0.00529	0.00898	0.01316	0.01759

approach is to set p small and adjust to UCL accordingly.

Figure 3 displays the average value of $\pi(\tau \leq T|D^T)$ and $\tilde{\pi}_T(\tau \leq T)$ for different values of K for the merging and windowing procedures, for 250 replications of 500 in-control profiles of length $n = 128$. The approximation accuracy is generally very good for the MWBB, and is very poor for windowing. For the remainder of this section, we will no longer consider the windowing procedure due to its poor performance. Next, using the same generated data, the middle 50% quantile of $\pi(\tau \leq T|D^T)$ for different procedures is displayed in Figure 4. We again notice that a better approximation is obtained for $p = 1/370$ rather than $p = 1/100$, with the approximation becoming worse the further we move beyond the ARL; we note that the approximation quality is less important for times far beyond the ARL since a false alarm will likely have occurred by this point regardless.

Lastly, Figure 5 displays the L_1 difference between $\pi(\tau \leq T|D_T)$ and $\tilde{\pi}_T(\tau \leq T)$. The error for the MWBB does accumulate for larger values of T, but is small enough to not be of concern even for $T \gg K_{\max}$ because the error is small relative to the UCL. We again see better performance for small values of p, with the approximation quality not increasing substantially for $p = 1/370$.

4.1.5. Out of control performance

We now introduce different types of functional changes into our generated profiles. We evaluate procedures according to their average detection delay, which we denote as ARL_1, and the probability of a false alarm. We also report the standard deviation of the detection delay as SDRL_1. We use the same values of ω, s, p, n, and number of replications as before.

We simulate data with change points at $\tau \in \{1, 10, 50\}$. These choices of τ allow us to assess the ability of each method to immediately signal a change and the ability to run in control before signaling a change.

Results for $\tau = 10$ are given in Tables 5 and 6, with results for other values of τ given in the Supplementary Material. We see that there is a modest increase in detection delay for smaller values of $K_{\max}$, particularly when the magnitude of the change m is small.

As p decreases (with ARL_0 fixed at $1/p$), the probability of a false alarm decreases sharply, with a modest increase in ARL_1. In the case of the LRT, M^*, and Windowing methods, the low probability of false alarms must be balanced against the substantially longer detection delays. For example, we can match the performance of Windowing at $\text{ARL}_0 = 100$ in terms of average run length and false positive rate with *better* performance of MWBB at $K_{\max} = 10$ even with a longer average run-length $\text{ARL}_0 = 370$. We conclude that, rather than using windowing to obtain a better false alarm rate, it is preferable to use FWBB with a higher ARL, which is better simultaneously in terms of ARL, detection delay, and false alarm probability. Overall, we observe that MWBB and FWBB have very similar performance characteristics, indicating that MWBB performs well as an approximation to FWBB.

4.2. Non-normal errors

To assess the sensitivity of our framework to the form of the error distribution, we replace the normal errors in (1) with a skewed error distribution. In the following experiments, we generate 250 in-control profiles and set $p = \frac{1}{100}$ and $\text{ARL}_0 = 100$, with $(\omega, s) = (0.05, 1.74)$. We consider skew normal errors; similar experiments with gamma-distributed errors are given in the Supplementary Material. We set ϵ_i^t to have a skew-normal distribution, with density $f(\epsilon) = (2/\gamma)\phi\{(\epsilon - \xi)/\gamma\}\ \Phi\{\alpha(\epsilon - \xi)/\gamma\}$, where $\phi(\epsilon)$ and $\Phi(\epsilon)$ are the standard normal density and distribution functions respectively. The parameter ξ is a location parameter, γ is a scale parameter, and α is a skewness parameter. After being generated, these errors are then scaled to have mean 0 and variance 1. A table giving the calibration and performance metrics of our procedure are given in Table S.8 in the Supplementary Material, with $(\xi, \gamma, \alpha) = (5, 3, 6)$.

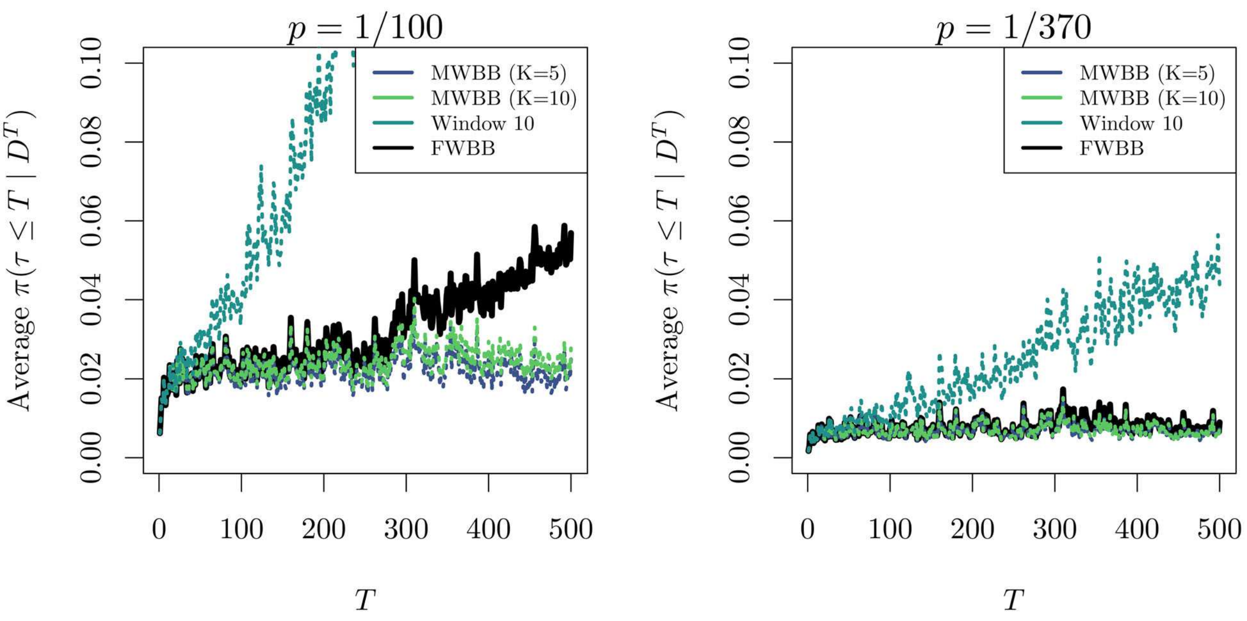

Figure 3. The average value of $\pi(\tau \leq T|D^T)$ and $\tilde{\pi}_T(\tau \leq T)$ at each T from the 250 replications of the in control setting used to calibrate each method under different settings of p.

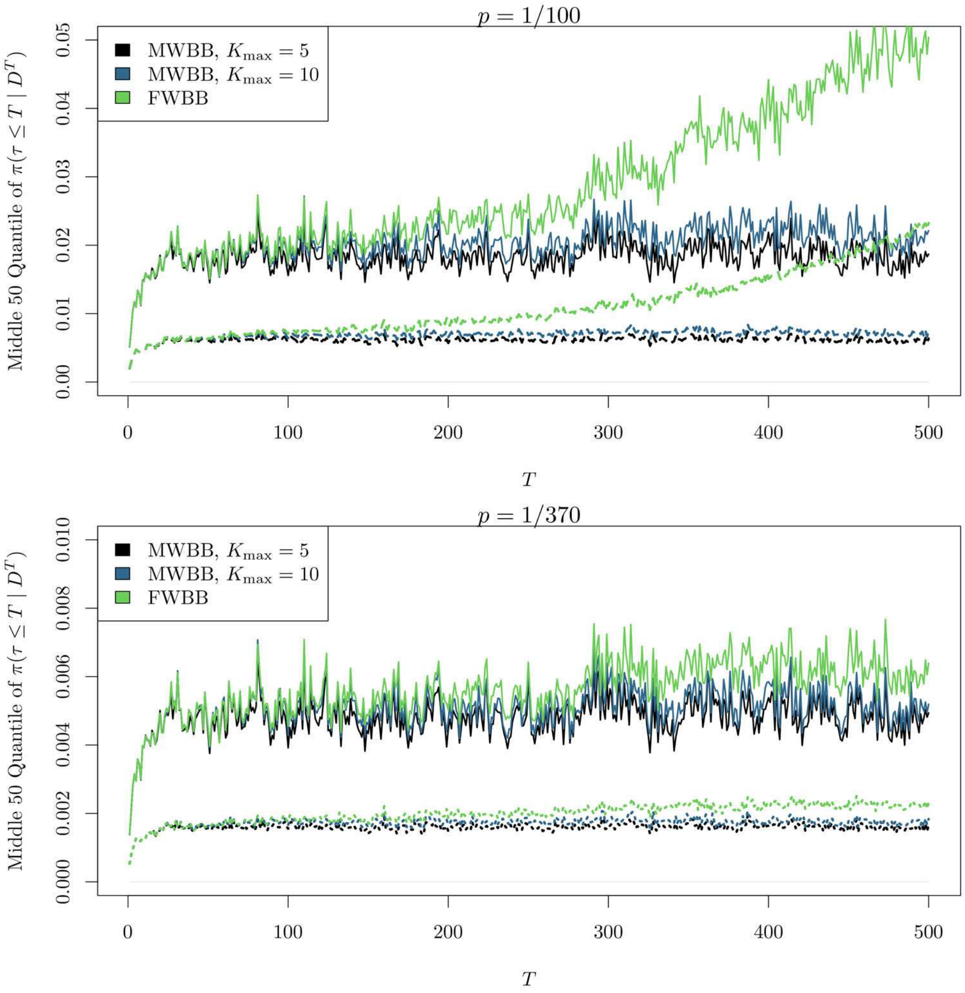

Figure 4. The middle 50 percent quantile of $\pi(\tau \leq T|D^T)$ of different procedures under different settings of p. Solid lines give the 75th percent quantile, while dashed lines give the 25th percent quantile.

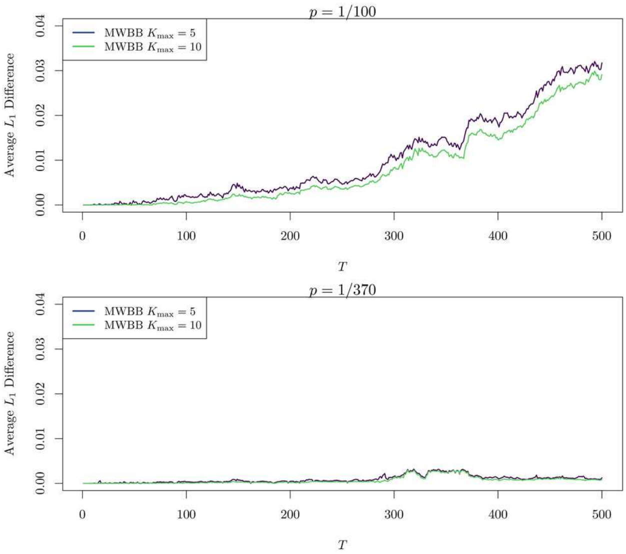

Figure 5. The average L_1 distance between $\pi(\tau \leq T|D_T)$ and $\tilde{\pi}_T(\tau \leq T)$ at each time under different settings of p,.

After calibrating the UCL, we investigate the ability to detect a change when the simulated profiles have the skew normal errors. Table 7 gives the detection delay and false alarm probabilities under this setting. We see a modest increase in detection delay from Table S.4, but the MWBB methodology is still capable of detecting change before the LRT method and the windowing procedures. For skew-normal errors, MWBB still obtains performance nearly equivalent to FWBB; $K_{\max} = 20,$ for example, is essentially identical to FWBB in terms of performance. We also investigated larger τ for this error setting and found similar results. The performance metrics can be found in Table S.9 in the Supplementary Material.

4.3. Vertical density profile data

We now analyze a dataset consisting of vertical density profiles (VDP) of pressed wood panels presented in Walker and Wright (2002). Understanding the VDP of these panels is important because the VDP is related to the machinability of the wood. The data consists of measurements of the density of panels under varying depths across the thickness of the boards. We observe $T = 24$ profiles, which were collected in three different 8 hour shifts (Shifts A, B, and C). Observations of each wood panel were taken at 314 depths. For simplicity, we trim the profiles on either end down to the next lowest dyadic profile length ($n = 256$). This data has been the analyzed in many other works in statistical process control. Operating in a similar framework to ours, Walker and Wright (2002) develop a class of generalized additive models that are used to assess the sources of variation within profiles, while Williams, Woodall, and Birch (2007) construct various T^2 statistics based on nonlinear models to determine control limits for monitoring.

The data are displayed in Figure 6. In order to apply our Phase II methodology to the dataset, we follow the approach for defining the in control set used in McGinnity, Chicken, and Pignatiello (2015), which performs three different analyses with Shift A, B, or C assumed to be in control. We take f^0 to be the mean of the in-control profiles. These profiles are displayed in Figure 6.

There is a large amount of variability in the mean VDP across profiles. As our primary goal is to detect differences in the shape of the profiles rather than in

Table 5. The (ARL_1, SDRL_1, PFA) for different values of $K_{\max}$ and different out of control conditions, based on 250 replications, $\tau = 10$ and $p = \frac{1}{100}$.

Method	m	Triangular	Parabolic	Broken Line	Local Jumps
MWBB	0.01	(25.45, 15.69, 0.06)	(13.18, 8.46, 0.07)	(17.32, 10.21, 0.05)	(34.22, 25.23, 0.06)
$K_{\max} = 5$	0.09	(3.20, 1.26, 0.08)	(2.00, 0.86, 0.05)	(2.57, 1.01, 0.06)	(4.42, 1.79, 0.04)
	0.25	(1.52, 0.53, 0.08)	(1.13, 0.35, 0.06)	(1.21, 0.42, 0.06)	(1.98, 0.71, 0.05)
MWBB	0.01	(26.23, 14.77, 0.08)	(11.40, 7.05, 0.08)	(16.84, 9.67, 0.07)	(32.70, 22.08, 0.06)
$K_{\max} = 10$	0.09	(3.25, 1.32, 0.06)	(1.95, 0.86, 0.06)	(2.56, 1.10, 0.08)	(4.59, 2.10, 0.08)
	0.25	(1.47, 0.53, 0.07)	(1.09, 0.29, 0.07)	(1.29, 0.46, 0.06)	(1.94, 0.68, 0.06)
MWBB	0.01	(24.08, 13.76, 0.10)	(12.34, 7.88, 0.05)	(16.56, 9.83, 0.04)	(34.42, 21.11, 0.07)
$K_{\max} = 20$	0.09	(3.42, 1.30, 0.05)	(2.09, 0.87, 0.06)	(2.47, 1.03, 0.07)	(4.65, 2.07, 0.05)
	0.25	(1.53, 0.51, 0.09)	(1.10, 0.30, 0.03)	(1.21, 0.43, 0.07)	(1.93, 0.66, 0.06)
FWBB	0.01	(23.90, 13.67, 0.05)	(11.37, 7.38, 0.10)	(15.58, 9.92, 0.08)	(33.08, 20.94, 0.05)
	0.09	(3.26, 1.27, 0.08)	(1.97, 0.79, 0.04)	(2.58, 1.15, 0.09)	(4.29, 2.04, 0.08)
	0.25	(1.50, 0.51, 0.09)	(1.11, 0.31, 0.06)	(1.28, 0.45, 0.08)	(1.88, 0.64, 0.09)
LRT	0.01	(94.26, 83.57, 0.07)	(99.45, 84.50, 0.06)	(90.55, 84.55, 0.08)	(100.50, 86.66, 0.08)
	0.09	(10.63, 9.14, 0.06)	(7.81, 6.41, 0.13)	(9.28, 7.52, 0.08)	(19.44, 15.60, 0.10)
	0.25	(1.75, 1.04, 0.08)	(1.35, 0.63, 0.09)	(1.62, 0.95, 0.06)	(3.78, 2.74, 0.09)
Window 10	0.01	(43.13, 31.61, 0.04)	(16.05, 12.16, 0.02)	(24.89, 16.03, 0.02)	(49.04, 33.99, 0.04)
	0.09	(3.64, 1.35, 0.02)	(2.31, 0.95, 0.03)	(2.85, 1.21, 0.02)	(4.89, 1.96, 0.02)
	0.25	(1.65, 0.60, 0.02)	(1.13, 0.33, 0.03)	(1.37, 0.49, 0.03)	(2.12, 0.66, 0.03)
$M*$	0.01	(85.84, 86.75, 0.09)	(86.11, 85.08, 0.09)	(82.53, 82.75, 0.072)	(85.74, 82.20, 0.07)
	0.09	(18.60, 17.91, 0.09)	(18.42, 18.70, 0.09)	(19.04, 18.00, 0.10)	(21.91, 23.41, 0.06)
	0.25	(3.40, 2.86, 0.10)	(3.36, 2.80, 0.11)	(3.47, 2.72, 0.12)	(3.41, 3.11, 0.06)

Table 6. The (ARL$_1$, SDRL$_1$, PFA) for different values of $K_{\max}$ and different out of control conditions, based on 250 replications, $\tau = 10$ and $p = \frac{1}{370}$.

Method	m	Triangular	Parabolic	Broken Line	Local Jumps
MWBB	0.01	(36.41, 19.75, 0.02)	(16.47, 8.93, 0.01)	(24.06 12.50, 0.02)	(58.61, 37.05, 0.03)
$K_{\max} = 5$	0.09	(3.66, 1.33, 0.01)	(2.35, 0.94, 0.03)	(3.18, 1.16, 0.01)	(5.52, 2.05, 0.02)
	0.25	(1.76, 0.51, 0.03)	(1.14, 0.35, 0.02)	(1.40, 0.49, 0.02)	(2.28, 0.73, 0.00)
MWBB	0.01	(32.19, 14.36, 0.01)	(16.83, 9.57, 0.03)	(24.04, 12.44, 0.02)	(50.38, 25.44, 0.02)
$K_{\max} = 10$	0.09	(3.75, 1.27, 0.03)	(2.37, 0.97, 0.01)	(3.05, 1.13, 0.02)	(5.33 2.07 0.02)
	0.25	(1.67, 0.55, 0.02)	(1.18, 0.39, 0.00)	(1.40, 0.50, 0.01)	(2.16 0.65 0.01)
MWBB	0.01	(31.45, 14.06, 0.02)	(16.46, 9.18, 0.01)	(22.33, 11.86, 0.01)	(48.19, 23.60, 0.01)
$K_{\max} = 20$	0.09	(3.87, 1.38, 0.01)	(2.30, 0.98, 0.02)	(3.08, 1.14, 0.02)	(5.45, 1.85, 0.02)
	0.25	(1.70, 0.57, 0.02)	(1.15, 0.36, 0.02)	(1.41, 0.51, 0.02)	(2.31, 0.70, 0.03)
FWBB	0.01	(33.92, 14.25, 0.01)	(17.12, 9.28 0.03)	(23.32, 11.53, 0.01)	(48.92, 24.52, 0.02)
	0.09	(3.87, 1.40, 0.02)	(2.52, 0.99, 0.02)	(3.07, 1.10, 0.02)	(5.43, 2.09, 0.02)
	0.25	(1.76, 0.59, 0.02)	(1.21, 0.41, 0.01)	(1.39, 0.50, 0.00)	(2.20, 0.75, 0.01)
LRT	0.01	(185.31, 108.13, 0.04)	(190.39, 105.31, 0.02)	(186.78, 108.03, 0.03)	(194.82, 102.33 0.02)
	0.09	(18.75, 14.10, 0.03)	(12.23, 9.60 0.02)	(17.69, 13.47, 0.03)	(37.89, 24.57, 0.01)
	0.25	(2.03, 1.22, 0.04)	(1.63, 1.05, 0.01)	(1.95, 1.17, 0.02)	(5.68, 4.46, 0.02)
Window 10	0.01	(156.43, 88.36, 0.00)	(46.08, 30.35, 0.00)	(87.02, 60.45, 0.00)	(206.91, 83.81, 0.00)
	0.09	(5.51, 1.77, 0.00)	(4.23, 1.26, 0.00)	(4.74, 1.35, 0.00)	(7.50, 2.19, 0.00)
	0.25	(3.54, 1.08, 0.00)	(2.27, 1.49, 0.00)	(2.76, 1.48, 0.00)	(3.89, 0.68, 0.00)
$M*$	0.01	(115.28, 115.63, 0.07)	(106.63, 97.86, 0.07)	(113.42, 119.68, 0.04)	(114.56, 121.75, 0.05)
	0.09	(22.20, 20.74, 0.07)	(21.48, 21.30, 0.06)	(22.09, 22.00, 0.07)	(24.63, 25.25, 0.05)
	0.25	(3.61, 2.97, 0.08)	(3.61, 3.19, 0.09)	(3.80, 2.90, 0.10)	(3.83, 3.57, 0.04)

Table 7. The ARL$_1$ (SDRL$_1$) for different values of $K_{\max}$ and different out of control conditions, based on 250 replications, $\tau = 1$ and $p = \frac{1}{100}$ and skew normal errors.

Method	m	Triangular	Parabolic	Broken Line	Local Jumps
MWBB	0.01	29.74 (16.55)	51.54 (39.57)	33.4 (19.03)	51.10 (37.23)
$K_{\max} = 5$	0.09	3.85 (1.51)	6.30 (2.62)	4.31 (1.70)	6.05 (2.44)
	0.25	1.67 (0.56)	2.58 (0.99)	1.89 (0.63)	2.42 (0.76)
MWBB	0.01	28.69 (14.81)	44.82 (28.35)	31.42 (16.24)	45.77 (29.48)
$K_{\max} = 10$	0.09	3.86 (1.51)	6.30 (2.63)	4.32 (1.70)	6.04 (2.43)
	0.25	1.67 (0.56)	2.58 (0.99)	1.89 (0.63)	2.42 (0.76)
MWBB	0.01	27.96 (14.41)	42.75 (25.48)	31.02 (15.76)	43.65 (26.98)
$K_{\max} = 20$	0.09	3.85 (1.51)	6.30 (2.62)	4.31 (1.70)	6.04 (2.43)
	0.25	1.67 (0.56)	2.58 (0.99)	1.89 (0.63)	2.42 (0.76)
FWBB	0.01	27.92 (14.37)	42.68 (25.25)	30.96 (15.69)	43.55 (26.76)
	0.09	3.85 (1.51)	6.30 (2.62)	4.31 (1.70)	6.04 (2.43)
	0.25	1.67 (0.56)	2.58 (0.99)	1.89 (0.63)	2.42 (0.76)
LRT	0.01	75.97 (67.72)	78.38 (70.90)	76.96 (69.27)	78.58 (73.04)
	0.09	15.31 (12.30)	45.49 (42.07)	22.40 (18.50)	27.91 (22.55)
	0.25	2.15 (1.30)	6.32 (5.04)	2.98 (2.03)	5.70 (4.18)
Window 10	0.01	52.02 (36.33)	73.24 (53.50)	56.12 (41.17)	71.25 (51.46)
	0.09	3.94 (1.50)	6.60 (2.78)	4.50 (1.75)	6.38 (2.50)
	0.25	1.72 (0.56)	2.69 (0.98)	1.95 (0.64)	2.49 (0.78)

their location on the y-axis, we center each profile to have mean 0 as a preprocessing step.

To determine an appropriate UCL, we calibrated each method under consideration by generating in control profiles by bootstrapping appropriately scaled errors from the assumed in control profiles. The bootstrapped errors were randomly selected deviations of the defined in control profiles from the estimated f^0, taken at each point within the profile length. The randomly selected error is then scaled by the standard deviation of the errors at each of the 256 points in the profiles. A more detailed description of this bootstrapping procedure can be found in McGinnity, Chicken, and Pignatiello (2015). Selection of hyperparameters is described in Section 2.4. In the results in Table 8, the UCL for the FWBB and the MWBB procedures are nearly identical, with MWBB accurately approximating FWBB. Hence, for the VDP data, FWBB and MWBB perform essentially the same when running in-control.

To illustrate the MWBB procedure's ability to run online in control for a long stretch of time, as well as quickly detect a change, we generated a sequence of 100 in control profiles from the defined in control profiles using the bootstrapping procedure described above, followed by profiles from shifts A, B and C. Results are given in Figure 7 for $K_{\max} = 5, 10, 20$ and FWBB.

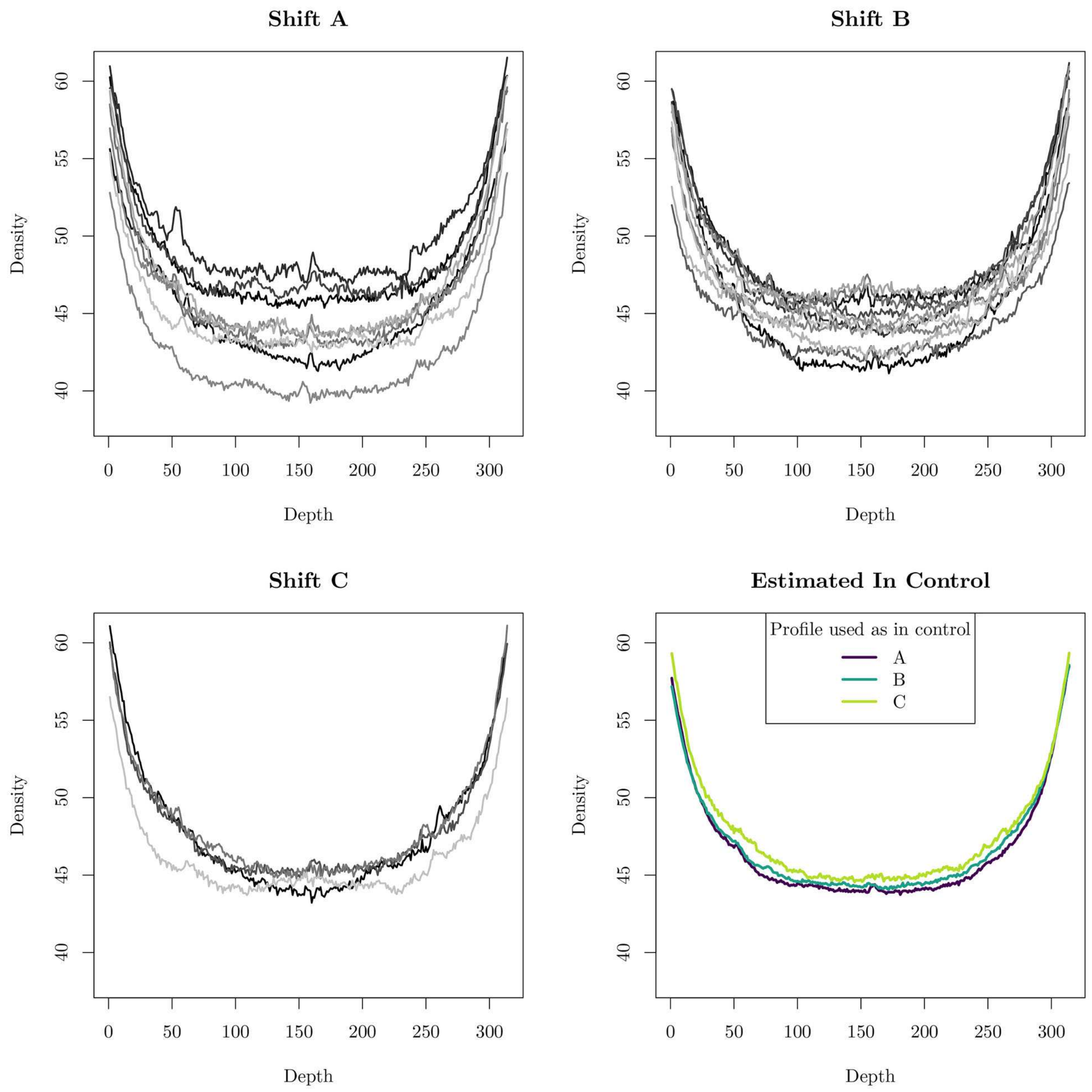

Figure 6. VDPs obtained during Shift A, B, and C, as well as the estimated in-control profile obtained from the three different shifts.

Table 8. Calibration summary and L_1 difference of MWBB from FWBB of the VDP data using 3 different sets of in control data.

Profiles in control	Method	UCL	ARL$_0$	SDRL$_0$	Average L_1 Difference
Shift A	$K_{\max} = 5$	0.17	100.06	68.46	0.000534
	$K_{\max} = 10$	0.17	100.47	69.57	0.000181
	$K_{\max} = 20$	0.16	100.14	69.57	0.000055
	Full	0.16	100.14	69.57	—
	LRT	0.02	100.43	67.49	—
Shift B	$K_{\max} = 5$	0.14	100.41	67.06	0.000607
	$K_{\max} = 10$	0.14	100.02	66.74	0.000181
	$K_{\max} = 20$	0.14	100.02	66.74	0.000053
	Full	0.14	100.02	66.74	—
	LRT	0.01	100.12	69.96	—
Shift C	$K_{\max} = 5$	0.07	100.27	66.37	0.000283
	$K_{\max} = 10$	0.07	101.29	67.47	0.000083
	$K_{\max} = 20$	0.07	101.29	67.47	0.000022
	Full	0.07	101.29	67.47	—
	LRT	0.002	100.70	74.19	—

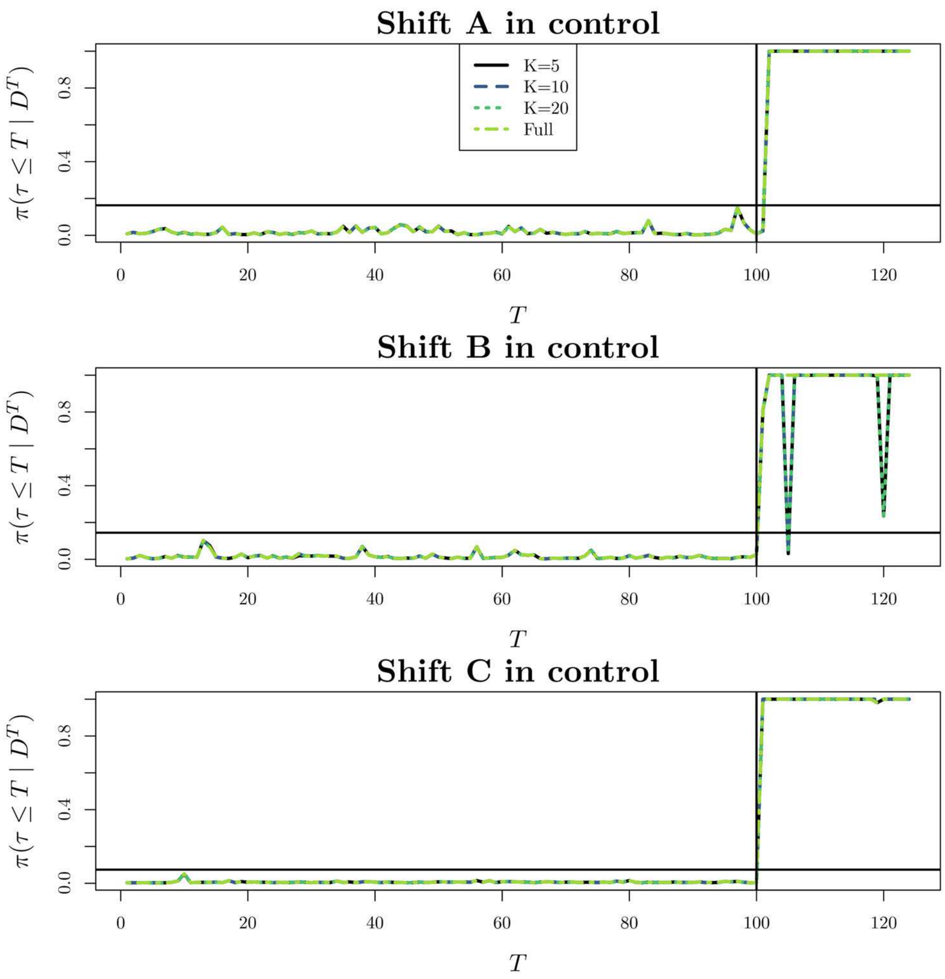

Figure 7. Probability of change of 100 bootstrapped in control profiles generated from the in control datasets. The vertical line reflects the time point at which profiles the VDP data are introduced. The horizontal line reflects the calibrated UCL for each in control set.

We see that the process is classified as in control for the first 100 profiles, and out of control when the profiles from the shifts are introduced. We also see that, when shift B is considered in control, the probability of a change jumps down when shift B profiles are introduced. We note that our assumption that each shift consists entirely of in-control profiles is contradicted by this experiment, as the second profile in shift A leads to the process being classified as out of control; we make the working assumption that each shift is in control with respect to itself, but a proper Phase I method is required in practice. To assess the impact of this assumption, we performed a permutation study to determine which profiles signal a change with high frequency.

Intuitively, profiles which deviate from in control behavior will have a high probability of signaling out of control, while profiles which behave similar to an in control process will rarely signal out of control.

Results are displayed in Figure 8. We see that which shift is in control has an effect on which profiles signal a change, with the in control shift having fewer out of control profiles.

In Figure 9, we repeated the simulation of Figure 7, but with the profiles classified as out of control in Shift A removed. In this study, the in control Shift A profiles no longer detect a change, while the out of control profiles from Shift A do (as do profiles from Shift B and Shift C).

The propensity for the wavelet-based approach to classify profiles as out of control is related to both (i) its ability to detect even subtle changes in profiles from the estimated in control behavior and (ii) the fact that we assume that the noise terms within each profile are independent, while auto-correlation is apparent in the VDP data. The independence assumption is used in works such as Kang and Albin (2000), Kim, Mahmoud, and Woodall (2003), and Williams, Woodall, and Birch (2007), but is dubious in this case; for wavelet based

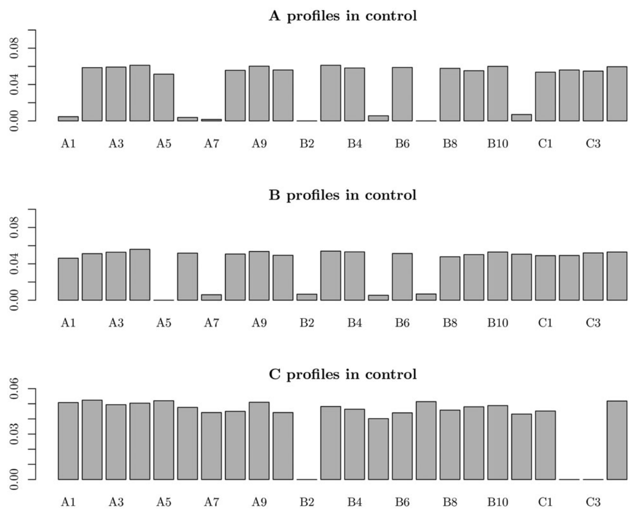

Figure 8. Results of the permutation study, displaying each shift against its probability of signaling out of control. As all sequences are eventually classified as out of control, the probabilities sum to 1.

methods, this assumption makes the methodology highly sensitive. A promising avenue for future work is accounting for auto-correlation in the data.

While our procedure is strictly a Phase II method, it is informative to compare the results of the permutation study with the results of other Phase I analyses to determine what features of a profile lead to it being classified as out of control. Figure 10 displays three profiles that signal a change with high frequency under the three different control settings, while Figure 11 displays three different profiles that do not signal a change. We see that in control behavior is associated with deviating from the in control profile by a fixed amount (due to the centering), whereas out of control behavior deviates from the overall shape of the in control profile. For instance, we find profile A2 to be out of control, and we see that this is because the

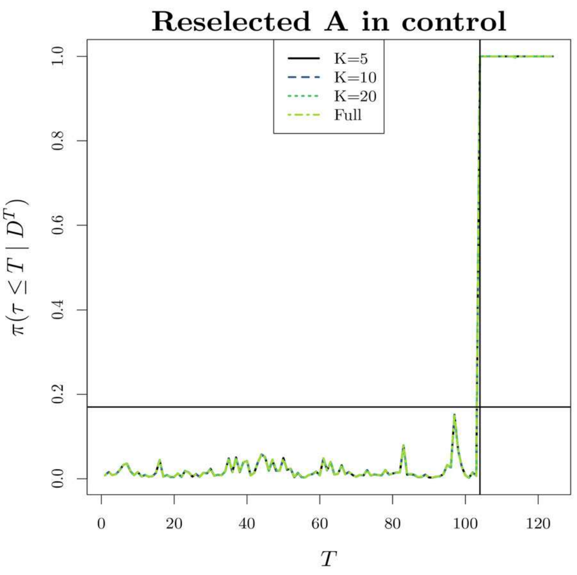

Figure 9. Probability of change of 100 bootstrapped in control profiles generated from the in control datasets. The vertical line reflects the time point at which profiles outside of our in control set are introduced. The horizontal line reflects the UCL.

VDP "bends" further toward the middle of the profile. By contrast, results from Williams, Woodall, and Birch (2007), which are given in Table 9, A2 is found to be in control.

5. Discussion and possible future work

Bayesian approaches to quickest change detection have a long history, but are difficult to apply with complex models due to the computational complexity of computing the posterior distribution. In this paper we addressed the problem of performing quickest change detection with a functional response by developing an accurate approximation to a Bayesian analysis. Our approach incurs a fixed computational cost

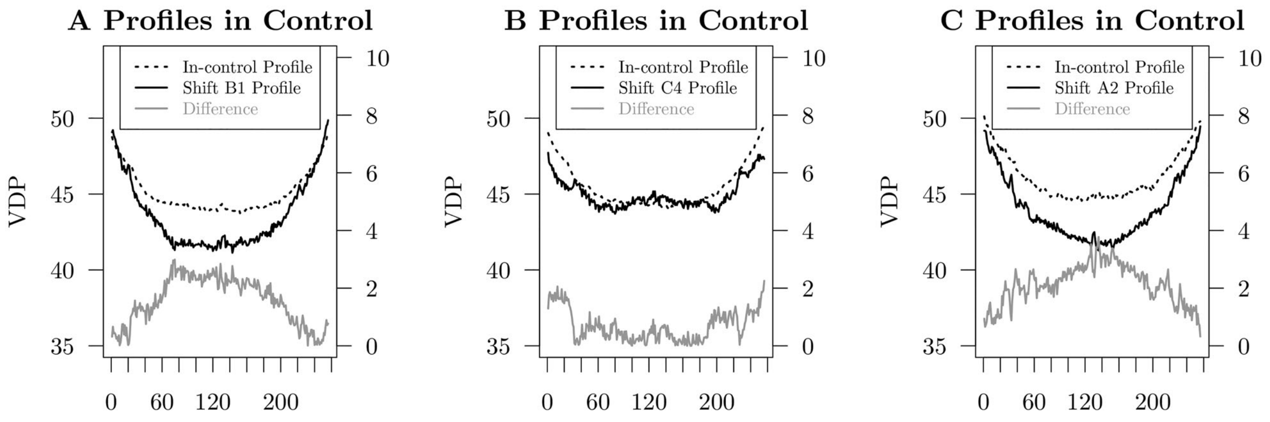

Figure 10. Example of profiles determined to be out of control by the permutation study under three different in control settings. The difference between the profile from the in-control mean is given in gray.

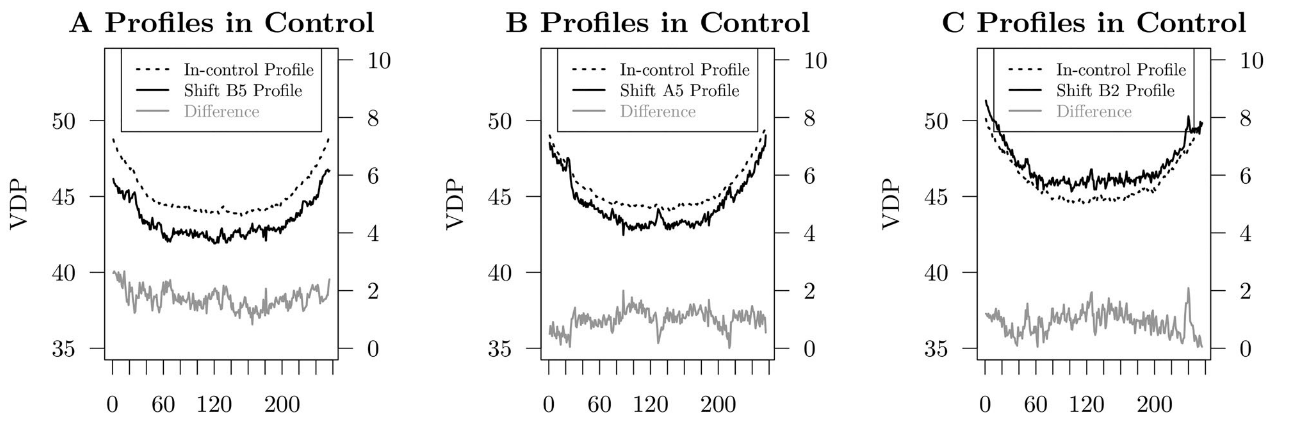

Figure 11. Example of profiles determined to be in control by the permutation study under three different in control settings. The difference between the profile from the in-control mean is given in gray.

Table 9. Classification of in control (IC) and out of control (OOC) using the method of Williams, Woodall, and Birch (2007).

A1	A2	A3	A4	A5	A6	A7	A8	A9	B1	B2	B3	B4	B5	B6	B7	B8	B9	B10	B11	C1	C2	C3	C4
IC	IC	IC	OOC	IC	IC	IC	IC	OOC	IC	IC	IC	IC	IC	OOC	IC	IC	OOC	IC	IC	IC	IC	IC	OOC

at each time while still allowing for complex functional deviations from in control to be detected.

There are several interesting areas for future research. Throughout, we have assumed that there is a single shift from in control to out of control, with the profiles being homogeneous within each condition. A related problem is to consider a profile which migrates out of control slowly, or allow for the process to return to in control after a certain amount of time.

Additionally, our methods are derived under the assumption of iid Gaussian errors or known non-normal errors, with the error level constant across profiles. For the VDP data, this assumption was seen to be suspect. Further work might consider unknown error distribution, or allow for the incorporation of dependent errors within a profile.

Disclosure statement

No potential conflict of interest was reported by the author(s).

Funding

This research was supported by the Office of the Secretary of Defense, Directorate of Operational Test and Evaluation, and the Test Resource Management Center under the Science of Test research program. Contract #FA8075-14-D-0019 / FA8075-18-F-1525. The views expressed in this article are those of the authors and do not reflect the official policy or position of the United States Air Force, Department of Defense, or the U.S. Government. This research was also supported by the National Science Foundation under awards DMS 1712870 and DMS-1810831.

About the authors

Wright Shamp is a Doctoral Candidate in Statistics at Florida State University. His research interests include statistical process control, nonparametric density estimation, and Bayesian estimation.

Roumen Varbanov is a statistician at HubSpot in Cambridge, MA. He currently works on the People Analytics team, which provides statistical consulting and analytical tools to answer workforce-related questions.

Eric Chicken is a Professor in the Department of Statistics at Florida State University. His research interests include statistical process control, nonparametric regression and density estimation, statistical estimation via wavelets, and Bayesian estimation.

Antonio Linero is an Assistant Professor in the Department of Statistics and Data Sciences at The University of Texas at Austin. His research broadly focuses on developing flexible Bayesian methods, including appropriate methods for complex longitudinal data and model selection tools within the Bayesian nonparametric framework for high dimensional problems.

Yun Yang is an Assistant Professor in the Department of Statistics at University of Illinois Urbana-Champaign. His research interests lie broadly in machine learning, scalable Bayes inference, and theoretical foundations of high dimensional problems.

References

Abramovich, F., T. Sapatinas, and B. W. Silverman. 1998. Wavelet thresholding via a Bayesian approach. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)* 60 (4):725–49. doi: 10.1111/1467-9868.00151.

Adams, R. P., and D. J. MacKay. 2007. Bayesian online changepoint detection. Technical Report. *arXiv Preprint arXiv:0710.3742*

Amiri, A.,. W. A. Jensen, and R. B. Kazemzadeh. 2009. A case study on monitoring polynomial profiles in the automotive industry. *Quality and Reliability Engineering International* 26 (5):509–20. doi: 10.1002/qre.1071.

Bishop, C. M. 2006. *Pattern recognition and machine learning*. New York: Springer Science & Business Media.

Blei, D. M., A. Kucukelbir, and J. D. McAuliffe. 2017. Variational inference: A review for statisticians. *Journal of the American Statistical Association* 112 (518):859–77. doi: 10.1080/01621459.2017.1285773.

Chicken, E., J. J. Pignatiello, and J. R. Simpson. 2009. Statistical process monitoring of nonlinear profiles using wavelets. *Journal of Quality Technology* 41 (2):198–212. doi: 10.1080/00224065.2009.11917773.

Colosimo, B. M., Q. Semeraro, and M. Pacella. 2008. Statistical process control for geometric specifications. *Journal of Quality Technology* 40 (1): 1–18.

Crouse, D. F., P. Willett, K. Pattipati, and L. Svensson. 2011. A look at Gaussian mixture reduction algorithms. In *Information Fusion (FUSION), 2011 Proceedings of the 14th International Conference on*, 1–8. IEEE.

Donoho, D. L., and J. M. Johnstone. 1994. Ideal spatial adaptation by wavelet shrinkage. *Biometrika* 81 (3): 425–55. doi: 10.1093/biomet/81.3.425.

Fan, J. 1996. Test of significance based on wavelet thresholding and Neyman's truncation. *Journal of the American Statistical Association* 91 (434):674–88. doi: 10.1080/01621459.1996.10476936.

Hawkins, D. M., and K. Zamba. 2005. Statistical process control for shifts in mean or variance using a changepoint formulation. *Technometrics* 47 (2):164–73. doi: 10.1198/004017004000000644.

Jeong, M. K., J.-C. Lu, and N. Wang. 2006. Wavelet-based SPC procedure for complicated functional data. *International Journal of Production Research* 44 (4): 729–44. doi: 10.1080/00207540500222647.

Jin, J., and J. Shi. 2001. Automatic feature extraction of waveform signals for in-process diagnostic performance improvement. *Journal of Intelligent Manufacturing* 12 (3): 257–68. doi: 10.1023/A:1011248925750.

Johnstone, I. M., and B. W. Silverman. 2005. Empirical Bayes selection of wavelet thresholds. *The Annals of Statistics* 33 (4):1700–52. doi: 10.1214/009053605000000345.

Kang, L., and S. L. Albin. 2000. On-line monitoring when the process yields a linear profile. *Journal of Quality Technology* 32 (4):418–26. doi: 10.1080/00224065.2000.11980027.

Kim, K.,. M. A. Mahmoud, and W. H. Woodall. 2003. On the monitoring of linear profiles. *Journal of Quality Technology* 35 (3):317–28. doi: 10.1080/00224065.2003.11980225.

Mallat, S. G. 1989a. Multiresolution approximations and wavelet orthonormal bases of l^2 (R). *Transactions of the American Mathematical Society* 315 (1):69–87. doi: 10.2307/2001373.

Mallat, S. G. 1989b. A theory for multiresolution signal decomposition: The wavelet representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 11 (7):674–93. doi: 10.1109/34.192463.

McGinnity, K., E. Chicken, and J. J. Pignatiello Jr. 2015. Nonparametric changepoint estimation for sequential nonlinear profile monitoring. *Quality and Reliability Engineering International* 31 (1):57–73. doi: 10.1002/qre.1657.

Ogden, T. 2012. *Essential wavelets for statistical applications and data analysis*. Boston: Birkhauser.

Runnalls, A. R. 2007. Kullback-Leibler approach to gaussian mixture reduction. *IEEE Transactions on Aerospace and Electronic Systems* 43 (3):989–99. doi: 10.1109/TAES.2007.4383588.

Schieferdecker, D., and M. F. Huber. 2009. Gaussian mixture reduction via clustering. In *12th International Conference on Information Fusion*, 1536–43. IEEE.

Shiryaev, A. N. 1963. On optimum methods in quickest detection problems. *Theory of Probability & Its Applications* 8 (1):22–46. doi: 10.1137/1108002.

Tartakovsky, A. G., and V. V. Veeravalli. 2005. General asymptotic Bayesian theory of quickest change detection. *Theory of Probability & Its Applications* 49 (3):458–97. doi: 10.1137/S0040585X97981202.

Varbanov, R., E. Chicken, A. Linero, and Y. Yang. 2019. A Bayesian approach to sequential monitoring of nonlinear profiles using wavelets. *Quality and Reliability Engineering International* 35 (3):761–75. doi: 10.1002/qre.2409.

Vidakovic, B. 2009. *Statistical modeling by wavelets, volume 503*. New York: John Wiley & Sons.

Walker, E., and S. P. Wright. 2002. Comparing curves using additive models. *Journal of Quality Technology* 34 (1): 118–29. doi: 10.1080/00224065.2002.11980134.

Williams, J. D., W. H. Woodall, and J. B. Birch. 2007. Statistical monitoring of nonlinear product and process quality profiles. *Quality and Reliability Engineering International* 23 (8):925–41. doi: 10.1002/qre.858.

Willsky, A., and H. Jones. 1976. A generalized likelihood ratio approach to the detection and estimation of jumps in linear systems. *IEEE Transactions on Automatic Control* 21 (1):108–12. doi: 10.1109/TAC.1976.1101146.

Winistorfer, P. M., T. M. Young, and E. Walker. 1996. Modeling and comparing vertical density profiles. *Wood and Fiber Science* 28 (1):133–41.