Statistica Sinica Preprint No: SS-2024-0182	
Title	Sequential Change Point Detection in High-dimensional Vector Auto-regressive Models
Manuscript ID	SS-2024-0182
URL	http://www.stat.sinica.edu.tw/statistica/
DOI	10.5705/ss.202024.0182
Complete List of Authors	Yuhan Tian and Abolfazl Safikhani
Corresponding Authors	Abolfazl Safikhani
E-mails	asafikha@gmu.edu
Notice: Accepted author version.	

Sequential Change Point Detection in High-dimensional Vector Auto-regressive Models

Yuhan Tian and Abolfazl Safikhani

University of Florida and George Mason University

Abstract: Sequential/Online change point detection involves continuously monitoring time series data and triggering an alarm when shifts in the data distribution are detected. We propose an algorithm for real-time identification of alterations in the transition matrices of high-dimensional vector auto-regressive models. This algorithm initially estimates transition matrices and error term variances using regularization techniques applied to training data, then computes a specific test statistic to detect changes in transition matrices as new data batches arrive. We establish the asymptotic normality of the test statistic under the scenario of no change points, subject to mild conditions. An alarm is raised when the calculated test statistic exceeds a predefined quantile of the standard normal distribution. We demonstrate that as the size of the change (jump size) increases, the test's power approaches one. Empirical validation of the algorithm's effectiveness is conducted across various simulation scenarios. Finally, we discuss two applications of the proposed methodology: analyzing shocks within S&P 500 data and detecting the timing of seizures in EEG data.

Key words and phrases: auto correlation; break point; sequential data; structural break; temporal dependence.

1. INTRODUCTION

Abrupt changes in daily life are often perceived as anomalies, typically requiring careful study to prevent future repercussions. In a data set, such abrupt changes are usually triggered by shifts in the data-generating process. Detecting these changes precisely and quickly is essential for understanding their origins and mitigating potential harm. Consequently, change point detection (CPD) has become a critical research area in both data science and statistics, with real-world applications spanning power systems, quality control, and advertising (Routtenberg and Xie, 2017; Page, 1954; Zhang et al., 2017). Most CPD methods, classified here as offline CPD, require access to the full dataset and aim to pinpoint change point locations accurately. However, with advancements in cloud storage and computing, streaming data has become ubiquitous, necessitating a different CPD approach for monitoring incomplete and dynamic data in real time. Online (or Sequential) CPD addresses this need by triggering alarms as changes are detected in data streams. A robust online CPD method should therefore achieve low false alarm rates, minimal detection delays, and efficient computational processing. In this study, we introduce an online change point detection algorithm that meets all these requirements.

A wide range of online CPD methods has been documented in the literature, primarily focusing on techniques to detect changes in distribution parameters of univariate

data, such as the mean, variance, or overall distribution. These methods include early applications of Shewhart charts, cumulative sum (CUSUM) charts, and exponentially weighted moving average (EWMA) charts for quality control (Shewhart, 1930; Page, 1954; Roberts, 2000), as well as more recent advances based on likelihood ratio tests (Hawkins and Zamba, 2005). For more comprehensive coverage of control charts for univariate and multivariate time series, see Montgomery (2019); Qiu (2013). Recent advances in computational power and data storage have enabled broader interest in multivariate time series, with applications across finance, weather forecasting, healthcare, and industrial operations. Developing online CPD algorithms for multivariate (or high-dimensional) data introduces two main challenges: adapting univariate test techniques for multivariate data and accounting for inter-component correlations, both contemporaneous and cross-correlated. These challenges complicate the theoretical guarantees for false alarm control and detection delay and require careful attention to computational efficiency in an online setting. Two common solutions include applying univariate CPD methods independently to each series or transforming the multivariate time series into a single series for univariate CPD analysis (Jandhyala et al., 2013). The former approach may weaken detection power by overlooking common change points, while the latter may be ineffective if change points in some series are masked by noise in the combined data. To address this, several methods aggregate component-wise test statistics rather than observations

(Chen et al., 2022; Gösmann et al., 2022; Bardwell et al., 2019; Xu et al., 2021). For instance, the algorithm in Bardwell et al. (2019) examines each series individually to generate a profile-likelihood-like statistic for change points and post-processes the results to identify common change points. However, this method lacks theoretical guarantees for false positives and detection delays. Another approach by Chen et al. (2022) uses likelihood ratio tests across scales and coordinates, but it assumes independent Gaussian observations, limiting its applicability to real-world data with temporal dependencies and cross-correlations. Additionally, Gösmann et al. (2022) integrates component-wise schemes using a maximum statistic, which converges to a Gumbel distribution as dimension and sample size increase, though this method only detects mean shifts and not changes in variance or covariance. Finally, Xu et al. (2021) introduces an online CPD approach with sampling control, selecting only a few observable series at each time point and employing a sequential probability ratio test. This approach reduces dimensionality by focusing on selected series, though it assumes independent and identically distributed samples. These methods, while capable of handling high-dimensional settings, struggle with contemporary and cross correlations inherent in multivariate time series due to their reliance on independent test statistics per series.

For online CPD in data with dependencies and cross-correlations, treating the multivariate series as an integrated entity is more viable. In Bayesian Online CPD (Adams

and MacKay, 2007), change point inference is based on the posterior distribution of the current run length, updated sequentially with new data. This method is flexible through its choice of predictive distribution, but it is not readily adaptable to high-dimensional data, where the likelihood becomes computationally infeasible as dimensions increase. Additionally, Bayesian Online CPD's time complexity scales linearly with the number of observations, making it unsuitable for long time series (in contrast, our algorithm's complexity is independent of the number of observations, depending only on window size). Several recent methods employ graph-based techniques for online CPD in high-dimensional data. For instance, k-nearest neighbor (k-NN) algorithms by Chen (2019); Chu and Chen (2022) perform two-sample tests on k-NN sequentially as data arrives. These algorithms apply to high-dimensional series with contemporaneous correlations, given a suitable similarity measure. However, temporal dependencies can undermine k-NN methods, as the local neighborhood's definition becomes unreliable over time. Specifically, the choice of neighbors may change as patterns shift, making it difficult to adapt effectively to temporal dependencies. Projection-based control charts offer a practical approach for handling high-dimensionality and correlations in process monitoring. A notable example is Zhang et al. (2020), which uses random projections to reduce dimensionality, creating subprocesses that are monitored by local nonparametric control charts and then fused for decision-making. PCA-based control charts provide another option for

dimension reduction, addressing various high-dimensional data types (De Ketelaere et al., 2015). For example, dynamic PCA-based charts (Ku et al., 1995) manage autocorrelation by including lagged data, while recursive PCA charts (Choi et al., 2006) handle nonstationarity by updating parameters with a forgetting factor, and moving window PCA charts (He and Yang, 2008) maintain a recent data window. However, projection-based methods can lack interpretability since identified changes may involve multiple variables, complicating the source identification. For a detailed overview of CPD methods, see Aminikhanghahi and Cook (2017). Despite the extensive work on online CPD, few methods effectively manage high-dimensional settings with cross correlations and even fewer offer theoretical guarantees, highlighting the need for our proposed algorithm tailored to these challenges.

To address high-dimensional data with temporal and cross correlations, our algorithm is based on the vector auto-regressive (VAR) model, represented in equation (2.1). The key parameters of a VAR model are its transition matrices, which capture temporal and cross dependencies among observations. This linear structure provides computational and analytical efficiency, making VAR a staple in multivariate time series analysis, with applications spanning economics Rosser Jr and Sheehan (1995), neuroscience Goebel et al. (2003), and quality control Pan and Jarrett (2012). Changes within a VAR time series manifest as alterations in its transition matrices, as described in Wang et al. (2019), enabling our algorithm to detect shifts in higher-order structures,

including temporal and cross correlations. This capability differentiates our approach, as most online CPD algorithms focus on changes in mean (Gösmann et al., 2022; Dette and Gösmann, 2020; Hawkins and Zamba, 2005; Aminikhanghahi et al., 2018), variance (Dette and Gösmann, 2020; Hawkins and Zamba, 2005; Aminikhanghahi et al., 2018), or contemporary correlations (Dette and Gösmann, 2020; Zhang et al., 2023; Cabrieto et al., 2017). With the rise of high-dimensional data, VAR models have become increasingly important. Despite their wide applications, a clear gap remains: to our knowledge, no online or sequential CPD algorithms are explicitly designed for high-dimensional VAR models. Our algorithm aims to bridge this gap by providing an online CPD approach for detecting abrupt changes in transition matrices in high-dimensional VAR models.

To explain our algorithm, we introduce two essential quantities: n and ω. Here, n represents the number of observations in our training data set (historical data set), carefully selected to exclude any change points—a common practice in online CPD research (Qiu and Xie, 2022; Gösmann et al., 2022; Chen et al., 2022). The parameter ω denotes the permissible detection delay, as data is observed incrementally in an online setting. To assess whether time t is a change point, a few subsequent data points are needed, referred to as the "pre-specified detection delay" (ω). Following Aminikhanghahi and Cook (2017), our algorithm is therefore a ω-real-time algorithm. Our algorithm has two primary steps. First, we estimate transition matrices and error

variances using the training data set, denoted as $X_1, X_2, \ldots, X_n$. Given the high-dimensional nature of the model, we use an ℓ_1-penalized least squares estimator, which needs to be computed only once for the entire monitoring process. Second, we compute a test statistic on sequential data batches of size ω, specifically $X_{t+1}, X_{t+2}, \ldots, X_{t+\omega}$, observed at times $t, t+1, \ldots$. Our method triggers an alarm if the test statistic exceeds a predefined threshold (see Section 3.2 for details). Additionally, we include a refinement step to locate change points and reduce false alarms (see Section 5). In Theorem 1, we establish the asymptotic normality of the test statistic under conditions without change points, and in Theorem 2, we examine the relationship between the power of the test and the jump size (the difference in model parameters before and after the change point). This analysis requires examining the fourth-order properties of the VAR time series, as parameter consistency in high-dimensional settings generally involves only first- and second-order properties (Basu and Michailidis, 2015; Wong et al., 2020). The proof is detailed in Section S3 of the Supplementary Materials. The normality of the test statistic allows for selecting a threshold to control the average run length (false alarm rate) without resorting to costly Monte Carlo simulations, a step often required by other online CPD algorithms (Chen et al., 2022; Qiu and Xie, 2022). This is particularly beneficial in high-dimensional online scenarios. Our algorithm demonstrates robust numerical performance, achieving well-controlled average run length and short detection delay, as detailed in Section 7. It also surpasses several

alternative methods in computational efficiency, an advantage supported by results in Sections 7 and 8. This speed is critical for online CPD applications. Additionally, the algorithm is resource-efficient, requiring only moderate memory and data storage for parameter estimates.

Our algorithm's main contributions are as follows. It is an online change point detection algorithm specifically designed for high-dimensional VAR models, addressing a notable gap in the field. It detects changes in higher-order structures, such as temporal and cross correlations, while handling high dimensionality and dependencies. The algorithm has theoretical guarantees, with asymptotic normality allowing control over the average run length. We also analyze the link between algorithm power and jump size. It optimizes resource usage, reducing the need for Monte Carlo simulations during threshold selection and enhancing the efficiency of the CPD process.

The paper is organized as follows: Section 2 describes the model setup and change point detection problem. The test statistic and detection algorithm are outlined in Section 3, and theoretical results are presented in Section 4. Section 5 introduces a refinement approach for precise change point estimation. Section 6 extends the algorithm to multiple change points. Performance results on synthetic data are presented in Section 7, while real data applications are discussed in Section 8. Finally, Section 9 provides concluding remarks and future research directions.

Notations: In this paper, when referring to a vector $v \in \mathbb{R}^p$, we denote its j-th feature as v_j. The ℓ_q norms are represented by $\|v\|_q = \left(\sum_{j=1}^p |v_j|^q\right)^{1/q}$, where $q > 0$. We use $\|v\|_0$ to denote $|\text{supp}(v)| = \sum_{i=1}^p \mathbf{1}[v_j \neq 0]$, and $\|v\|_\infty$ to represent $\max_j |v_j|$. For an indexed vector, $v_i \in \mathbb{R}^p$ for $i = 1, \ldots, n$, its j-th feature is denoted as $v_{i,j}$. In the case of a matrix A, $\rho(A)$, $\|A\|$, and $\|A\|_F$ denote its spectral radius $|\Lambda_{\max}(A)|$, operator norm $\sqrt{\Lambda_{\max}(A'A)}$, and Frobenius norm $\sqrt{\text{tr}(A'A)}$, respectively. Additionally, $\|A\|_{\max}$, $\|A\|_\infty$, and $\|A\|_1$ denote the coordinate-wise maximum (in absolute value), maximum absolute row sum, and maximum absolute column sum of matrix A, respectively. For matrix A, its maximum and minimum eigenvalues are represented as $\Lambda_{\min}(A)$ and $\Lambda_{\max}(A)$, and the element in the i-th row and j-th column is denoted as $a_{i,j}$. The i-th unit vector in $\mathbb{R}^p$ is indicated as e_i. Throughout this paper, the notation $A \gtrsim B$ signifies the existence of an absolute constant c, independent of model parameters, such that $A \geq cB$. The notation $A \asymp B$ is used to denote $A \gtrsim B$ and $B \gtrsim A$. The notation $\Gamma_X(\ell)$ is used to represent $\text{Cov}(X_t, X_{t+l})$. A' and A^* stand for the ordinary transpose and the Hermitian transpose of matrix A respectively. Convergence in probability and in distribution is indicated by $\xrightarrow{p}$ and $\xrightarrow{D}$ respectively. The variance of a random variable or vector x is denoted as $\text{VAR}(x)$.

2. MODEL FORMULATION

We assume the p-dimensional data are generated from a finite order VAR(h) process with the transition matrices $A_1, \ldots, A_h$ before the occurrence of the change. After the change point, i.e., from time $t^* + 1$, transition matrices are changed to $A_1^*, \ldots, A_h^*$ with $(A_1^*, \ldots, A_h^*) \neq (A_1, \ldots, A_h)$. The lag h may vary before and after the change point. In such situations, we refer to h as the maximum of the two lags, and we augment the process with the smaller lag by including a few zero-matrix transition matrices. Consequently, to simplify matters, we assume that the lag h remains constant both before and after the change point without loss of generality. Formally, data points before the change point, denoted as $\{\ldots, X_{t^*-1}, X_{t^*}\}$ (including the training set $\{X_{-h+1}, \ldots, X_n\}$), and data points subsequent to the change point, denoted as $\{X_{t^*+1}, X_{t^*+2}, \ldots\}$, are generated according to the following equations:

$$X_t = \sum_{l=1}^{h} A_l X_{t-l} + \varepsilon_t, \text{ for } t \leq t^*; \; X_t = \sum_{l=1}^{h} A_l^* X_{t-l} + \varepsilon_t, \text{ for } t > t^*. \tag{2.1}$$

Here, the error vectors ε_t are temporally independent, possessing a mean of zero and a covariance matrix denoted as $\Sigma = \sigma^2 I_p$. We refer to Remark 1 for a more flexible structure for the covariance of the error term. The VAR model naturally entails a high-dimensional parameter space, as the number of parameters scales quadratically with respect to data dimension (i.e. the number of parameters is of order p^2 where p is the data dimension). Even when p is relatively modest, this scaling leads to a

substantial parameter count, exposing the model to the challenges of high-dimensional settings.

3. DETECTION ALGORITHM

In this section, we provide details of the proposed detection algorithm. The algorithm consists of two main steps. We assume we have access to $n+h$ training data points in which there are no changes in transition matrices. In the first step, these data points are used to estimate the baseline transition matrices and variance of error terms. In the second step, new batches of observations of size ω are observed, and test statistics are computed using these batches and model parameter estimations from the first step. Large values of the test statistic indicate a potential change point.

3.1 Step I: Estimation of Transition Matrices and Error Variance

In this step, we aim to estimate the transition matrices and the variance of error terms using the provided training data. To achieve this, we construct a regression problem based on the training data denoted as $\mathbb{X}_{hist} = \{X_{-h+1}, \ldots, X_n\}$. This regression problem takes the following form:

$$\underbrace{\begin{bmatrix} X_n' \\ \vdots \\ X_1' \end{bmatrix}}_{\mathcal{Y}_n} = \underbrace{\begin{bmatrix} X_{n-1}' & \cdots & X_{n-h}' \\ \vdots & \ddots & \vdots \\ X_0' & \cdots & X_{-h+1}' \end{bmatrix}}_{\mathcal{X}_n} \underbrace{\begin{bmatrix} A_1' \\ \vdots \\ A_h' \end{bmatrix}}_{B^*} + \underbrace{\begin{bmatrix} \varepsilon_n' \\ \vdots \\ \varepsilon_1' \end{bmatrix}}_{E_n}.$$

This problem can be expressed in vector form as: $\text{vec}(\mathcal{Y}_n) = \text{vec}\left(\mathcal{X}_n B^*\right) + \text{vec}(E_n) = \left(I \otimes \mathcal{X}_n\right) \text{vec}\left(B^*\right) + \text{vec}(E_n)$. Alternatively, it can be represented as:

$$\underbrace{Y_n}_{np\times 1} = \underbrace{Z_n}_{np\times hp^2} \underbrace{\beta^*}_{hp^2\times 1} + \underbrace{\text{vec}(E_n)}_{np\times 1}.$$

We employ an ℓ_1-penalized least squares approach to estimate the transition matrices $A_1, A_2, \ldots, A_h$, which is equivalent to estimating β^*. Simultaneously, we estimate σ^2 and $\text{Var}(\varepsilon_{1,1}^2)$ using the method of moments:

$$\hat{\beta}_n = \underset{\beta\in\mathbb{R}^{hp^2}}{\text{argmin}} \left(\frac{1}{n}\|Y_n - Z_n\beta\|_2^2 + \lambda_n\|\beta\|_1\right), \tag{3.2}$$

$$\hat{\sigma}_n^2 = \frac{1}{pn}\sum_{i=1}^{n}\left\|\left(\sum_{l=1}^{h}\hat{A}_l X_{i-l}\right) - X_i\right\|_2^2, \tag{3.3}$$

$$\text{and } \hat{V}_n = \left|\frac{1}{pn}\sum_{i=1}^{n}\left\|\left(\sum_{l=1}^{h}\hat{A}_l X_{i-l}\right) - X_i\right\|_4^4 - \hat{\sigma}_n^4\right|. \tag{3.4}$$

The estimator for the transition matrices, employing ℓ_1-penalized least squares, exhibits several valuable properties, including consistency in high-dimensional settings (Basu and Michailidis, 2015). The parameter λ_n acts as a tuning parameter, controlling sparsity in the estimation. The choice of the tuning parameter λ_n is determined through cross-validation, and additional information can be found in the R package "sparsevar," as introduced in Vazzoler (2021). Finally, if the lag h for the VAR process is unknown, We recommend estimating it by comparing the Bayes information criterion (BIC), defined as $\text{BIC}(h) = \ln|\hat{\Sigma}(h)| + \frac{\ln n}{n}hp^2$, where $\hat{\Sigma}(h) = n^{-1}\sum_{t=1}^{n}\hat{\varepsilon}_{t,h}\hat{\varepsilon}_{t,h}'$. Here,

$\hat{\varepsilon}_{t,h}$ represents the residual at time t when a VAR(h) model is employed. To determine the appropriate lag, one should calculate BIC(h) for a grid of VAR models with various potential lags, using the historical data. The lag with the lowest BIC(h) should be selected as the estimated lag.

3.2 Step II: Test Statistic

Given the parameter estimates $\hat{\beta}_n$, $\hat{\sigma}_n^2$, $\hat{V}_n$, and the new observations $\mathbb{X}_{obs} = X_{t+1}, \ldots, X_{t+\omega}$ with $t > n - \omega$, we define the test statistic as follows:

$$\hat{T}_t^{(n,\omega)} = \sqrt{\frac{p\omega}{\hat{V}_n}} \left(\frac{\hat{R}_t^{(n,\omega)}}{p} - \hat{\sigma}_n^2 \right), \tag{3.5}$$

$$\text{where } \hat{R}_t^{(n,\omega)} = \frac{1}{\omega} \sum_{i=t+1}^{t+\omega} \left\| \left(\sum_{l=1}^{h} \hat{A}_l X_{i-l} \right) - X_i \right\|_2^2. \tag{3.6}$$

Finally, we compute the test statistic $\hat{T}_t^{(n,\omega)}$ for $t = n-\omega+1, n-\omega+2, \ldots$. An alarm will be raised at time $\hat{t}$ if $\left| \hat{T}_{\hat{t}}^{(n,\omega)} \right| > \Phi(1-\alpha/2)$, where $\Phi(\cdot)$ represents the standard normal quantile function.

The underlying concept behind the developed test statistic lies in its behavior under different scenarios. When no change points are present, our test statistic becomes a normalized sum of independent and identically distributed random variables, assuming our estimation of transition matrices is consistent. In such cases, the distribution of the test statistic closely approximates a standard normal distribution, as demonstrated in Theorem 1. Conversely, if a change point exists before time t, our test statistic

Algorithm 1 VAR_cpDetect_Online

Require: $data \in \mathbf{R}^{p\times T}$, n, ω, α, h

$t \leftarrow 0$; $\mathbb{X}_{hist} \leftarrow data[, t+1 : t+n+h]$; $\hat{\beta}_n \leftarrow \underset{\beta\in\mathbb{R}^{hp^2}}{\operatorname{argmin}} \left(\frac{1}{n}\left\|Y_n - Z_n\beta\right\|_2^2 + \lambda_n \left\|\beta\right\|_1\right)$

$\hat{\sigma}_n^2 \leftarrow \frac{1}{pn}\sum_{i=1}^{n}\left\|\left(\sum_{l=1}^{h}\hat{A}_l X_{i-l}\right) - X_i\right\|_2^2$;

$\hat{V}_n \leftarrow \left|\frac{1}{pn}\sum_{i=1}^{n}\left\|\left(\sum_{l=1}^{h}\hat{A}_l X_{i-l}\right) - X_i\right\|_4^4 - \hat{\sigma}_n^4\right|$; $t \leftarrow n+h-\omega+1$

while $t \le T-\omega$ **do**

$\mathbb{X}_{obs} \leftarrow data[, t-h+1 : t+\omega]$; $\hat{T}_t^{(n,\omega)} \leftarrow \sqrt{\frac{p\omega}{\hat{V}_n}}\left(\frac{\hat{R}_t^{(n,\omega)}}{p} - \hat{\sigma}_n^2\right)$

if $\left|\hat{T}_t^{(n,\omega)}\right| > \Phi(1-\alpha/2)$ **then**

raise alarm; $t \leftarrow t+1$

else

$t \leftarrow t+1$

end if

end while

exhibits a shift, as described in Theorem 2. Consequently, an alarm will be raised when $\left|\hat{T}_t^{(n,\omega)}\right| > \Phi(1-\alpha/2)$. The selection of ω is discussed in detail in Section S4.3 of the supplementary material. Identifying which components experience shifts after a raised alarm is crucial in high-dimensional settings. We propose using an online debiasing technique (Deshpande et al., 2023) to construct confidence intervals for the differences in transition matrices before and after the change to infer the changing

components; further details are provided in Section S7 of the supplementary material.

Remark 1. It is possible to extend the proposed algorithm to accommodate scenarios with variance heterogeneity. In such cases, the modified test statistic is defined as:

$$\hat{T}_t^{(n,\omega)} = \frac{\sqrt{\omega}\left(\hat{R}_t^{(n,\omega)} - \sum_{j=1}^{p}\hat{\sigma}_{n,j}^2\right)}{\sqrt{\sum_{j=1}^{p}\hat{V}_{n,j}}},$$

where $\hat{\sigma}_{n,j}^2$ and $\hat{V}_{n,j}$ are estimated separately using the method of moments for each component j. In order to keep the exposition of proposed methodology clear, we focus on fixed/homogeneous variance case in the remainder of the paper while the satisfactory performance under heterogeneous case is empirically illustrated in Section S4.6 in supplementary material. Note that extending the algorithm to scenarios with non-diagonal covariance matrices is discussed in Section 9.

4. THEORETICAL PROPERTIES

In this section, we present two theorems concerning the asymptotic behavior of the test statistic in two distinct scenarios: one when there are no change points, and the other when a change point exists. To derive these theorems, it is necessary to make the following assumptions.

Assumption 1. *The transition matrices exhibit sparsity, meaning that the vector β^* possesses a sparsity level denoted as s, represented as $\|\beta^*\|_0 = s$.*

Assumption 2. *The error terms, denoted as ε_t, are independent sub-Gaussian random vectors with a mean of zero and a variance of $\sigma^2 I_p$. Moreover, their sub-Gaussian*

norm is bounded by a constant K.

Assumption 3. *The VAR process is stable and stationary.*

Assumption 4. *The spectral density function, denoted as* $f_X(\theta) := \frac{1}{2\pi}\sum_{\ell=-\infty}^{\infty}\Gamma_X(\ell)e^{-i\ell\theta}$, *exists for* θ *within the interval* $[-\pi,\pi]$. *Additionally, its maximum and minimum eigenvalues are bounded on this interval, that is,*

$$\mathcal{M}\left(f_X\right) := \sup_{\theta\in[-\pi,\pi]}\Lambda_{\max}\left(f_X(\theta)\right) < \infty \text{ and } \mathfrak{m}\left(f_X\right) := \inf_{\theta\in[-\pi,\pi]}\Lambda_{\min}\left(f_X(\theta)\right) > 0.$$

Assumption 1 is common in high-dimensional models and plays a important role in addressing dimensionality issues. On the other hand, Assumption 2 is employed to manage the tail behavior of the data distribution. It's worth noting that the sub-Gaussian assumption can be relaxed to accommodate heavier-tailed distributions, such as the sub-Weibull distribution, albeit at the expense of a larger sample size requirement, as discussed in Wong et al. (2020). Assumption 3 is common in the time series literature, as seen in references like Lütkepohl (2005), and it ensures the existence of a unique stationary solution for the auto-regressive equations (2.1). Finally, Assumption 4 is essential for verifying the restricted eigenvalue and deviation bound conditions, as outlined in Loh and Wainwright (2012) and Basu and Michailidis (2015). These two properties are crucial prerequisites for establishing the consistency of the ℓ_1-regularized estimates in (3.2).

Theorem 1. Suppose that there are no change points in the data generation process,

and Assumptions 1-4 are satisfied. Then, with $\omega = o(n)$, $\omega \gtrsim s(\log h + 2\log p)$, $\frac{s(\log h+2\log p)}{\sqrt{p}} = o(\sqrt{n})$, $n^{1/2-a} \gtrsim \frac{\sqrt{s}}{p^{1/2-a}}$ for some $a \in (0, 1/2)$ and $n^{1/4-b} \gtrsim \frac{\sqrt{s}}{p^{3/4-b}}$ for some $b \in (0, 1/4)$, we have

$$\hat{T}_t^{(n,\omega)} \xrightarrow{D} \mathcal{N}(0,1) \text{ as } n \to \infty,$$

where $\mathcal{N}(0,1)$ represents the standard normal distribution.

This theorem forms the foundation of the proposed online detection algorithm by analyzing the marginal distribution of the test statistic in scenarios without change points. The asymptotic normality provides an objective basis for selecting the alarm threshold by utilizing the quantile function of the standard normal distribution. Consequently, the algorithm's average run length (ARL) can be controlled through the selection of the threshold α. Note that the dependence arising from overlapping windows may impact the ARL or false alarm rate of the algorithm, and thus, proper choice of the threshold α should be provided. As summarized in Section S4.1 of the supplementary material, the proposed method of selecting α (which does not account for potential overlapping dependence) has proven sufficient to control the ARL and maintain the false alarm rate. Typically, when the historical data set is sufficiently large for accurate parameter estimation, the average run length will be lower bounded by $1/\alpha$. The presence of a change in the model parameters will be indicated by significant deviations of the test statistic beyond a chosen quantile of the standard

normal distribution. It is important to mention that the sample size requirements outlined in Theorem 1 are relatively lenient, allowing for the consideration of high-dimensional scenarios, provided that the transition matrices are sparse. For example, when dealing with a fixed lag h, it is possible to select values such as $p = n^c$ and $\omega = (\log n \log p)^{1+\epsilon}$, where c and ϵ are positive constants. This choice remains valid as long as the sparsity level s satisfies $s = o\left((\log n)^{1+\epsilon} (\log p)^{\epsilon}\right)$.

Theorem 2. Assume the existence of a change point at time t^*, and assume that Assumptions 1-4 hold for the data both before and after this change point. Under the same conditions in Theorem 1 with additional conditions that $s(\log h + 2\log p) = o(\omega)$, $\sqrt{\frac{s(\log h+2\log p)}{\omega}} = o\left(\|\beta^* - \beta_{new}\|_2\right)$ and $\omega^\eta p^\eta \sqrt{\frac{s^3(\log h+2\log p)}{n}} = o\left(\|\beta^* - \beta_{new}\|_2\right)$ for some $\eta \in (0, 1/4)$, we have

$$\mathrm{P}(Z_{t^*+h}^{(n,\omega)} + \frac{c_l}{\sqrt{\hat{V}_n}} \sqrt{\frac{\omega}{p}} \|\beta^* - \beta_{new}\|_2^2 + L_{t^*+h}^{(n,\omega)} \leq \hat{T}_{t^*+h}^{(n,\omega)}$$
$$\leq Z_{t^*+h}^{(n,\omega)} + \frac{c_u}{\sqrt{\hat{V}_n}} \sqrt{\frac{\omega}{p}} \|\beta^* - \beta_{new}\|_2^2 + \left(L_{t^*+h}^{(n,\omega)}\right)') \geq 1 - \epsilon_{n,p,\omega},$$

where

$$Z_{t^*+h}^{(n,\omega)} \xrightarrow{D} \mathcal{N}(0,1) \text{ as } n \to \infty, \hat{V}_n \xrightarrow{p} \mathrm{Var}(\varepsilon_{1,1}^2) \text{ as } n \to \infty,$$

$$L_{t^*+h}^{(n,\omega)} = o_p\left(\sqrt{\frac{\omega}{p}} \|\beta^* - \beta_{new}\|_2^2\right), \left(L_{t^*+h}^{(n,\omega)}\right)' = o_p\left(\sqrt{\frac{\omega}{p}} \|\beta^* - \beta_{new}\|_2^2\right) \text{ and}$$

$$\epsilon_{n,p,\omega} = c_1 \exp(-c_2\omega) + c_3 \exp\left[-c_4(\log h + 2\log p)\right]$$

$$+ c_5 \exp\left(-c_6 n\right) + 2\exp(-c_7 p^\eta \omega^\eta + \log(\omega p h)),$$

for some positive constants $c_1, \ldots, c_7, c_l, c_u$ where β^* and β_{new} denote the vectorized

transition matrices before and after the change point, respectively.

Remark 2. The conditions introduced, which relate to the jump size denoted as $\|\beta^* - \beta_{new}\|_2$, are fundamental for evaluating the power of our test. Similar conditions are commonly found in the literature on change point detection and have been employed in various studies, such as assumption A3 in Safikhani and Shojaie (2022) and H2 in Chan et al. (2014). It is important to highlight that the flexibility of selecting a small value for η ensures that these conditions remain valid even in high-dimensional scenarios, as the term p^η can be controlled.

This theorem sheds light on the behavior of the test statistic in the presence of a change point. With a large sample size, our test statistic will have a lower bound that corresponds to a right-shifted standard normal distribution with high probability. The extent of this shift is influenced by the jump size. Furthermore, if we define $\tilde{t}$ as the last observation when the alarm is correctly raised $\left(\text{i.e., } \min\left\{t > t^* - \omega : \left|\hat{T}_t^{(n,\omega)}\right| > \Phi(1-\alpha/2)\right\} + \omega\right)$, we can establish the following corollary.

Corollary 2.1. Under the same conditions as outlined in Theorem 2, for any $k > 0$ and a sufficiently large n, there exists $\epsilon^{(n)} = o(1)$. If $\frac{c_l}{2\sqrt{\text{Var}(\varepsilon_{1,1}^2)}}\sqrt{\frac{\omega}{p}}\|\beta^* - \beta_{new}\|_2^2 > \Phi(1-\alpha/2) + k$, then the following inequalities hold:

$$P\left(\tilde{t} - t^* \leq \omega + h\right) \geq P\left(\left|\hat{T}_{t^*+h}^{(n,\omega)}\right| > \Phi(1-\alpha/2)\right) \geq 1 - \epsilon_{n,p,\omega} - \epsilon^{(n)} - \exp(-k^2/2).$$

As demonstrated in this corollary, when dealing with a substantial sample size and a considerable jump size, the detection delay $\tilde{t} - t^*$ is likely to be upper bounded by $\omega + h$. Furthermore, the power of our test, denoted as $P\left(\left|\hat{T}^{n,\omega}_{t^*+h}\right| > \Phi(1-\alpha/2)\right)$, can approach one as the jump size increases and the sample size grows. The proof for Corollary 2.1 relies on Theorem 2 and the concentration inequality for the standard normal distribution.

5. CHANGE POINT LOCALIZATION

Currently, our algorithm can only trigger an alarm when it detects change points within the observations contained in a window of size ω. However, determining the precise location of the change point remains unresolved. This situation is commonly encountered in the literature, see e.g. Chen et al. (2022); Mei (2010); Xie and Siegmund (2013); Chan (2017). Due to the limited number of observations available after the change point, accurately pinpointing its exact location proves to be challenging. Consequently, we propose a potential solution to refine the estimated location produced by our algorithm. The idea involves re-executing our algorithm using a smaller pre-specified detection delay of ω' within the current data window of size ω after an alarm is triggered. More specifically, upon the alarm being triggered by our algorithm at time $\hat{t}$, we treat the observations from $\hat{t}+1$ to $\hat{t}+\omega$ as the new data set that needs to be monitored. We will then apply our algorithm to this data set, employing a reduced

pre-specified detection delay of ω', and record the resulting estimated location of the change point as the refined estimate $(\hat{\hat{t}})$. In this scenario, our theorems remain valid if ω' satisfies the conditions for ω (for example, one can set $\omega' = \omega/5$). Figure 1 provides a visual demonstration of this process.

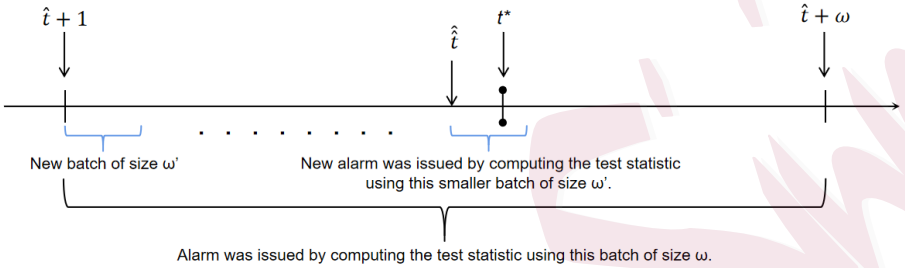

Figure 1: Illustration of Refinement

As illustrated in Figure 1, it is likely that the true change point falls within ω observations from the time the alarm is raised. This situation is the most frequent when the jump size is sufficiently large, as corroborated by Corollary 2.1. The refinement algorithm aims to reduce localization error in such cases. Another scenario occurs when a false alarm is raised. As the true change point has not been reached in this situation, the refinement process has a probability of not triggering any alarms within the data window. In such cases, we can consider the refinement process as a confirmation step. If no alarm is raised during the refinement process, we can conclude that the previous alarm was a false alarm. We can then ignore it and continue running the algorithm. This approach reduces the probability of raising false alarms, which is particularly valuable when false alarms are costly in practical

applications. Simulation D in Section S4.4 empirically demonstrates the effectiveness of the proposed refinement process. Details on the selection of ω' are provided in Section S4.4 of the supplementary material.

6. MULTIPLE CHANGE POINT SCENARIO

In this section, we consider the multiple change points case in which between change points, data is generated by stable and stationary VAR processes with different transition matrices and sub-Gaussian errors. Formally, if we have a sequence of true change points $\{t_0^* = 0, t_1^*, \ldots, t_{m-1}^*, t_m^* = T\}$, we have that

$$X_t = \sum_{l=1}^{h} A_l^{(j)} X_{t-l} + \varepsilon_t \text{ for } t_{j-1}^* < t \leq t_j^*, \tag{6.7}$$

where, without loss of generality, we assume that the VAR processes between change points have the same order h (otherwise, maximum of all lags will be selected as h) and the transition matrices between consecutive change points are different (i.e., $\{A_1^{(j)}, \ldots, A_h^{(j)}\} \neq \{A_1^{(j+1)}, \ldots, A_h^{(j+1)}\}$). Error terms ε_t are independent zero mean sub-Gaussian random vectors with variance $\sigma_{(j)}^2 I_p$ for $t_{j-1}^* < t \leq t_j^*$. Our Algorithm 1 can be implemented sequentially to address the proposed detection scenario, with the added assumption that the distance between change points is at least of the order $s\left(\log(hp^2)\right)$. This requirement ensures a sufficient number of observations are available before the next change point, allowing accurate parameter estimation for monitoring. Previous theoretical results still holds under the same assumptions, if

this new assumption is satisfied. This minimum distance condition is common in change point detection literature, see e.g. similar condition in Section 4.1 of Safikhani and Shojaie (2022) (see also Safikhani et al. (2022)). The implementation of this sequential detection algorithm is briefly discussed as follows. Once a change point is detected, a new training period is initiated to estimate the new transition matrices and variances. Subsequently, the monitoring period will be based on these new estimations, as illustrated in Figure 2. However, false alarms can be particularly costly under this implementation since they may trigger a training period that contains a change point. This situation not only leads to a missed detection but also contaminates the estimations for the upcoming monitoring. To effectively address this issue and reduce the probability of false alarms, we suggest applying the confirmation step, which was introduced in the refinement process in Section 5. It is also recommended to choose a conservative (small) value for α. The satisfactory performance of the sequential detection algorithm is confirmed empirically through synthetic data in Section S4.5.

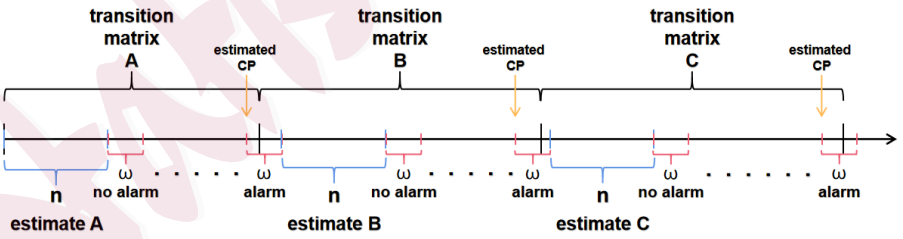

Figure 2: Implementation of detection algorithm with multiple change points

7. NUMERICAL COMPARISON

Due to space constraints, the assessment of the proposed algorithm with simulated data is presented in Section S4 of the supplementary material. Section S4 includes analyses of average run length, detection delay, window size selection, refinement effectiveness, performance under multiple change point scenarios, and robustness to variance heterogeneity, time-varying transition matrices, and non-sparse transition matrices. In this section, we compare the empirical detection performance of our method with baseline methods: gstream, ocp, TSL, ocd, Mei, XS and Chan. The gstream method, proposed in Chu and Chen (2022), utilizes a k-nearest neighbor approach to sequentially detect change points. The implementation of this algorithm is provided by the authors in Chen and Chu (2019). The Bayesian online change point detection algorithm, proposed in Adams and MacKay (2007), is implemented in the R package "ocp" Pagotto (2019). The TSL algorithm, introduced in Qiu and Xie (2022), is a non-parametric approach for online change point detection in multivariate time series data. The algorithm is implemented in Fortran by the authors, and we use their provided function for threshold selection. The algorithms, namely ocd, Mei, XS, and Chan introduced in Chen et al. (2022); Mei (2010); Xie and Siegmund (2013); Chan (2017), respectively, are designed to detect changes in multivariate time series data observed sequentially. They utilize likelihood ratio tests in individual coordinates and aggregate the resulting statistics across scales and coordinates. These algorithms

are available in the R package "ocd." We calculate the threshold for each algorithm using Monte Carlo simulation, as outlined in Section 4.1 of Chen et al. (2022).

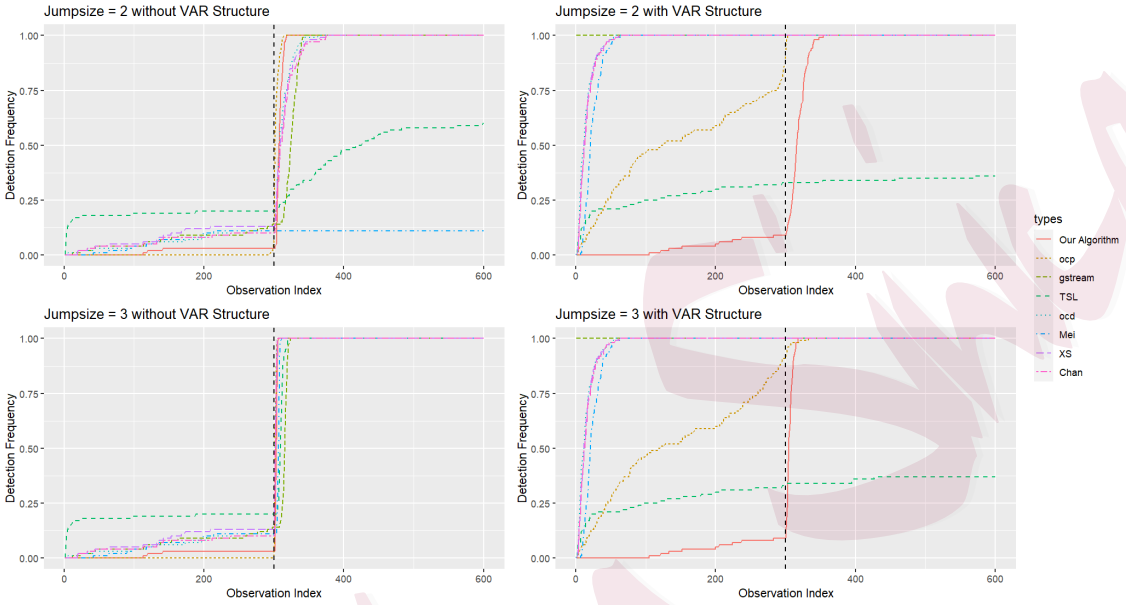

Figure 3: Comparison: These plots provide a summary of the detection frequency for all algorithms. The black dashed vertical line represents the location of the true change point. A perfect algorithm would have a detection frequency of zero before the line and reach one immediately after the line.

In the comparison, we apply our algorithm with $n = 500$, $\omega = 50$, and $\alpha = 1/1000$, along with the refinement and confirmation processes using a refine size of $1/10$. To ensure a fair comparison, we align most of the hyperparameters for the baseline methods with our choices. For example, we set $L = N0 = 500$ for gstream to match

Table 1: Comparison: The average execution times for each algorithm are listed based on simulations conducted on data sets of varying lengths, where the initial 500 data points are considered as the historical data set.

	our algorithm	ocp	gstream	TSL	ocd	Mei	XS	Chan
Length=3500	0.11s	18.26s	141.59s	46.58s	1.84s	0.68s	1.34s	1.33s
Length=4500	0.12s	31.63s	189.47s	66.28s	2.48s	0.86s	1.75s	1.74s
Length=5500	0.14s	48.06s	237.06s	90.95s	3.08s	1.11s	2.18s	2.21s

our choice of n, and we use $ARL = 1000$ and $\alpha = 1/1000$ for gstream. Similarly, we set $\lambda = 1000$ for ocp and $targetARL = 1000$ for TSL to match our choice of α. The target average run length is also set to 1000 for ocd, Mei, XS, and Chan. For the remaining hyperparameters, we either use the recommendations by authors or perform a grid search under the same experimental settings, selecting the hyperparameters that yield the best performance.

Since baseline algorithms are not specifically designed for data generated by a VAR process, we consider two scenarios in order to ensure a fair comparison. In the first scenario, data before the change point is generated from a constant mean model with independent and identically distributed errors (i.e., the transition matrix used is a zero matrix). In the second scenario, data before the change point is generated from a

VAR process with a transition matrix of $0.8 \times I_p$. In both scenarios, the data after the change point is generated by a new VAR process with a different transition matrix. We vary the jump size between the old and new transition matrices to compare the algorithm's sensitivity. In each repetition, we generate a data sequence with a length of 1100 and a dimension of 5 (due to space constraints, the case with $p = 100$ is provided in Section S4.7 of the supplementary material). The first 500 observations are used as historical data, and a change point is located at 800. After the first alarm is raised in each repetition, we terminate the algorithm and record the location of the last data point read at that time as $\tilde{t}$ (excluding historical data). We then construct an array consisting of $\tilde{t}$ zeros followed by $600 - \tilde{t}$ ones. This process is repeated for 100 repetitions, and we average the resulting arrays to obtain the detection frequency for each algorithm. A desirable algorithm should have a detection frequency of zero before 300 and a detection frequency of one after 300. The results are included in the Figure 3.

In the left figures, when data is generated without a VAR structure, our algorithm performs comparably to the ocp algorithm and outperforms the other baseline methods. Our algorithm maintains a low false alarm rate (small detection frequency before 300) and quickly detects the change (detection frequency reaches one shortly after 300). However, when we add the VAR structure to the data, all baseline methods suffer from correlations and are unable to maintain the same low false alarm rate as

before, as shown in the right figures. In contrast, our algorithm maintains similar performance as before, with only a slight increase in false alarm rate and detection delay.

Furthermore, we perform a brief simulation to assess the execution times of each algorithm. The setup and hyperparameter choices remain consistent with those in the comparison. We ensure that all algorithms continuously monitor the entire dataset with various sizes without halting even when an alarm is triggered, and we record the execution times. This procedure is repeated ten times to calculate the average execution times, and the outcomes are summarized in Table 1.

8. REAL DATA EXPERIMENTS

We evaluate the effectiveness of our approach (VAR_cpDetect_Online) and contrast its performance with competing methods in two real-world scenarios: S&P 500 data and EEG data. TSL, ocd, Mei, XS, and Chan are not suitable for this experimental setups as attempting to use them yielded unsatisfactory results, so we have excluded their outcomes from this section. The results for EEG data are deferred to Section S5.2 in the supplementary material to save space.

8.1 S&P 500 Data

This dataset consists of adjusted daily closing prices for 186 stocks in the S&P 500 from 2004-02-06 to 2016-03-02. Since closing prices are non-stationary, we apply the

data cleansing approach from Keshavarz et al. (2020) to compute daily log returns for each stock, yielding a dataset with 186 columns (stocks) and 3037 rows (trading days). The first 200 data points (up to 2004-11-22) are designated as historical data. For our method, we set $\omega = 22$ to match the typical number of trading days in a month and $\alpha = 1/5000$. Hyperparameters for competing algorithms were selected similarly, as described in Section 7. All methods were applied continuously to the entire dataset, without pausing when alarms were triggered. The locations of triggered alarms are documented in Section S5.1 of the supplementary material to save space. The aim of this experiment is to detect abnormal states indicated by data deviations from the baseline period. Applying the multiple change point detection approach discussed in Section 6 is unsuitable here, as the distance between change points does not meet required assumptions. Instead, we estimate the starting points of alarm clusters. Consecutive alarms within a window of ω observations are grouped into the same cluster, as they often stem from overlapping data segments. This suggests a high probability of a shared underlying abnormality. For each cluster, the onset is estimated using the refinement method in Section 5. Specifically, if an alarm's distance from the previous alarm exceeds ω, it is treated as a new cluster's start, and the refinement procedure is used to estimate the change point. If not, the alarm is considered part of the existing cluster, and change point estimation is not performed. The proposed algorithm raised alarms that formed 13 distinct clusters, with the

estimated starting points of these clusters treated as change points, as shown in Table 2. This table also highlights the historical events likely influencing the S&P500 index during these periods. In a previous study (Keshavarz et al., 2020), four alarm clusters were identified. Our algorithm aligns with these findings, with the starting points of these clusters indicated by an asterisk (*) in the table. Additionally, our algorithm identifies further periods of deviation from the baseline in the S&P 500 index. The average detection delay for our algorithm is 10.6, below the pre-specified threshold of $\omega = 22$, meaning the algorithm triggers alarms after approximately 10 additional observations to mark the start of each cluster. The ocp method missed the change points identified in the previous study in October 2007 and December 2010, while the gstream method missed the change point in August 2014. Moreover, the gstream method raised excessive alarms, covering about 54.7% of trading days, which limits its practicality for monitoring abnormal behavior. The execution times for the experiments were 50.63 seconds for our method, 227.64 seconds for ocp, and 50.03 seconds for gstream.

9. CONCLUDING REMARKS

In this paper, we introduced an online change point detection algorithm specifically designed for high-dimensional VAR models. This algorithm can effectively detect changes in higher-order structures, such as cross correlations. The algorithm's test

Date	Possible Real-World Event
2005-10-17	Concerns about the housing bubble and economic slowdown.
2006-07-13	Fed hints at pausing rate hikes amid inflation and housing worries.
2007-07-19	Early signs of the subprime mortgage crisis.
2007-10-12*	U.S. housing market declines significantly.
2010-01-14	Concerns over slow recovery and Eurozone debt crisis.
2010-04-20	BP oil spill raises environmental and economic concerns.
2010-12-22*	Strong holiday sales, but Eurozone concerns linger.
2011-07-26*	U.S. debt ceiling crisis and potential government default.
2012-05-24	Eurozone debt crisis, fears of Greece exiting the euro.
2013-12-20	Fed announces tapering of bond-buying program.
2014-08-21*	Market turbulence from geopolitical tensions and growth concerns.
2015-08-14	China devalues its currency, sparking global slowdown fears.
2015-12-10	Rising volatility ahead of expected Fed rate hike.

Table 2: Real-world events corresponding to changes in the S&P500 index.

statistic utilizes one-step-ahead prediction errors over a moving window of data. We demonstrated the asymptotic normality of our proposed test statistic under relatively mild conditions, which can encompass high-dimensional scenarios where

the number of parameters exceeds the sample size. Furthermore, we showed that the test's power approaches one with an increase in the jump size, and this was corroborated through numerical experiments. With respect to time complexity, we empirically demonstrated that our algorithm has a shorter computation time compared to competing algorithms. Our algorithm is currently tailored for data generated by VAR models with independent errors. Expanding its applicability to diverse error term structures is a challenging avenue for future research, given potential identifiability issues arising from general covariance structures. Further exploration includes relaxing some assumptions, such as substituting the sub-Gaussian distribution assumption on error terms with a more heavy-tail distribution assumption such as the sub-Weibull distribution. Investigating alternative forms of transition matrices, such as low rank or low rank combined with sparse structures Basu et al. (2019); Bai et al. (2023), is another promising area for research. Integrating the sequential updating technique (briefly discussed in Section S6 of the supplementary material) to improve transition matrix estimation when no alarm has been raised is an important yet challenging area for future research.

SUPPLEMENTARY MATERIALS

The online supplementary material includes proofs of lemmas and theorems, along with additional simulation and real data experiments.

References

Adams, R. P. and D. J. MacKay (2007). Bayesian online changepoint detection. *arXiv preprint arXiv:0710.3742*.

Aminikhanghahi, S. and D. J. Cook (2017). A survey of methods for time series change point detection. *Knowledge and information systems 51*(2), 339–367.

Aminikhanghahi, S., T. Wang, and D. J. Cook (2018). Real-time change point detection with application to smart home time series data. *IEEE Transactions on Knowledge and Data Engineering 31*(5), 1010–1023.

Bai, P., A. Safikhani, and G. Michailidis (2023). Multiple change point detection in reduced rank high dimensional vector autoregressive models. *Journal of the American Statistical Association*, 1–17.

Bardwell, L., P. Fearnhead, I. A. Eckley, S. Smith, and M. Spott (2019). Most recent changepoint detection in panel data. *Technometrics 61*(1), 88–98.

Basu, S., X. Li, and G. Michailidis (2019). Low rank and structured modeling of high-dimensional vector autoregressions. *IEEE Transactions on Signal Processing 67*(5), 1207–1222.

Basu, S. and G. Michailidis (2015). Regularized estimation in sparse high-dimensional time series models. *The Annals of Statistics 43*(4), 1535–1567.

Cabrieto, J., F. Tuerlinckx, P. Kuppens, M. Grassmann, and E. Ceulemans (2017). Detecting correlation changes in multivariate time series: A comparison of four non-parametric change point detection methods. *Behavior research methods 49*, 988–1005.

Chan, H. P. (2017). Optimal sequential detection in multi-stream data. *Annals of Statistics*.

Chan, N. H., C. Y. Yau, and R.-M. Zhang (2014). Group lasso for structural break time series. *Journal of the American Statistical Association 109*(506), 590–599.

Chen, H. (2019). Sequential change-point detection based on nearest neighbors. *The Annals of Statistics 47*(3), 1381–1407.

Chen, H. and L. Chu (2019). *gStream: Graph-Based Sequential Change-Point Detection for Streaming Data*. R package version 0.2.0.

Chen, Y., T. Wang, and R. J. Samworth (2022). High-dimensional, multiscale online changepoint detection. *Journal of the Royal Statistical Society Series B: Statistical Methodology 84*(1), 234–266.

Choi, S. W., E. B. Martin, A. J. Morris, and I.-B. Lee (2006). Adaptive multivariate statistical process control for monitoring time-varying processes. *Industrial & engineering chemistry research 45*(9), 3108–3118.

Chu, L. and H. Chen (2022). Sequential change-point detection for high-dimensional and non-euclidean data. *IEEE Transactions on Signal Processing 70*, 4498–4511.

De Ketelaere, B., M. Hubert, and E. Schmitt (2015). Overview of pca-based statistical process-monitoring methods for time-dependent, high-dimensional data. *Journal of Quality Technology 47*(4), 318–335.

Deshpande, Y., A. Javanmard, and M. Mehrabi (2023). Online debiasing for adaptively collected high-dimensional data with applications to time series analysis. *Journal of the American Statistical Association 118*(542), 1126–1139.

Dette, H. and J. Gösmann (2020). A likelihood ratio approach to sequential change point detection for a general class of parameters. *Journal of the American Statistical Association 115*(531), 1361–1377.

Goebel, R., A. Roebroeck, D.-S. Kim, and E. Formisano (2003). Investigating directed cortical interactions in time-resolved fmri data using vector autoregressive modeling and granger causality mapping. *Magnetic resonance imaging 21*(10), 1251–1261.

Gösmann, J., C. Stoehr, J. Heiny, and H. Dette (2022). Sequential change point detection in high dimensional time series. *Electronic Journal of Statistics 16*(1), 3608–3671.

Hawkins, D. M. and K. Zamba (2005). Statistical process control for shifts in mean or variance using a changepoint formulation. *Technometrics 47*(2), 164–173.

He, X. B. and Y. P. Yang (2008). Variable mwpca for adaptive process monitoring. *Industrial & Engineering Chemistry Research 47*(2), 419–427.

Jandhyala, V., S. Fotopoulos, I. MacNeill, and P. Liu (2013). Inference for single and multiple change-points in time series. *Journal of Time Series Analysis 34*(4), 423–446.

Keshavarz, H., G. Michailidis, and Y. Atchadé (2020). Sequential change-point detection in high-dimensional gaussian graphical models. *The Journal of Machine Learning Research 21*(1), 3125–3181.

Ku, W., R. H. Storer, and C. Georgakis (1995). Disturbance detection and isolation by dynamic principal component analysis. *Chemometrics and intelligent laboratory systems 30*(1), 179–196.

Loh, P.-L. and M. J. Wainwright (2012). High-dimensional regression with noisy and missing data: Provable guarantees with nonconvexity. *The Annals of Statistics 40*(3), 1637–1664.

Lütkepohl, H. (2005). *New introduction to multiple time series analysis*. Springer Science & Business Media.

Mei, Y. (2010). Efficient scalable schemes for monitoring a large number of data streams. *Biometrika 97*(2), 419–433.

Montgomery, D. C. (2019). *Introduction to statistical quality control*. John wiley & sons.

Page, E. S. (1954). Continuous inspection schemes. *Biometrika 41*(1/2), 100–115.

Pagotto, A. (2019). *ocp: Bayesian Online Changepoint Detection*. R package version 0.1.1.

Pan, X. and J. E. Jarrett (2012). Why and how to use vector autoregressive models for quality control: the guideline and procedures. *Quality & Quantity 46*(3), 935–948.

Qiu, P. (2013). *Introduction to statistical process control*. CRC press.

Qiu, P. and X. Xie (2022). Transparent sequential learning for statistical process control of serially correlated data. *Technometrics 64*(4), 487–501.

Roberts, S. (2000). Control chart tests based on geometric moving averages. *Technometrics 42*(1), 97–101.

Rosser Jr, J. B. and R. G. Sheehan (1995). A vector autoregressive model of the saudi arabian economy. *Journal of Economics and Business 47*(1), 79–90.

Routtenberg, T. and Y. Xie (2017). Pmu-based online change-point detection of imbalance in three-phase power systems. In *2017 IEEE Power & Energy Society Innovative Smart Grid Technologies Conference (ISGT)*, pp. 1–5. IEEE.

Safikhani, A., Y. Bai, and G. Michailidis (2022). Fast and scalable algorithm for detection of structural breaks in big var models. *Journal of Computational and Graphical Statistics 31*(1), 176–189.

Safikhani, A. and A. Shojaie (2022). Joint structural break detection and parameter estimation in high-dimensional nonstationary var models. *Journal of the American Statistical Association 117*(537), 251–264.

Shewhart, W. A. (1930). Economic quality control of manufactured product 1. *Bell System Technical Journal 9*(2), 364–389.

Vazzoler, S. (2021). *sparsevar: Sparse VAR/VECM Models Estimation*. R package version 0.1.0.

Wang, D., Y. Yu, A. Rinaldo, and R. Willett (2019). Localizing changes in high-dimensional vector autoregressive processes. *arXiv preprint arXiv:1909.06359*.

Wong, K. C., Z. Li, and A. Tewari (2020). Lasso guarantees for β-mixing heavy-tailed time series. *The Annals of Statistics 48*(2), 1124–1142.

Xie, Y. and D. Siegmund (2013). Sequential multi-sensor change-point detection. In *2013 Information Theory and Applications Workshop (ITA)*, pp. 1–20. IEEE.

Xu, Q., Y. Mei, and G. V. Moustakides (2021). Optimum multi-stream sequential change-point detection with sampling control. *IEEE Transactions on Information Theory 67*(11), 7627–7636.

Zhang, C., N. Chen, and J. Wu (2020). Spatial rank-based high-dimensional monitoring through random projection. *Journal of Quality Technology 52*(2), 111–127.

Zhang, J., Z. Wei, Z. Yan, M. Zhou, and A. Pani (2017). Online change-point detection in sparse time series with application to online advertising. *IEEE Transactions on Systems, Man, and Cybernetics: Systems 49*(6), 1141–1151.

Zhang, M., L. Xie, and Y. Xie (2023). Spectral cusum for online network structure change detection. *IEEE Transactions on Information Theory*.

Department of Statistics, University of Florida, Gainesville, FL 32611

E-mail: (yuhan.tian@ufl.edu)

Department of Statistics, George Mason University, Fairfax, VA 22030

E-mail: (asafikha@gmu.edu)