Partially Observable Multi-Sensor Sequential Change Detection: A Combinatorial Multi-Armed Bandit Approach

Chen Zhang
Department of Industrial Engineering
Tsinghua University
Beijing, China, 100084
zhangchen01@tsinghua.edu.cn

Steven C.H. Hoi
School of Information Systems
Singapore Management University
Singapore 178902
chhoi@smu.edu.sg

Abstract

This paper explores machine learning to address a problem of Partially Observable Multi-sensor Sequential Change Detection (POMSCD), where only a subset of sensors can be observed to monitor a target system for change-point detection at each online learning round. In contrast to traditional Multi-sensor Sequential Change Detection tasks where all the sensors are observable, POMSCD is much more challenging because the learner not only needs to detect on-the-fly whether a change occurs based on partially observed multi-sensor data streams, but also needs to cleverly choose a subset of informative sensors to be observed in the next learning round, in order to maximize the overall sequential change detection performance. In this paper, we present the first online learning study to tackle POMSCD in a systemic and rigorous way. Our approach has twofold novelties: (i) we attempt to detect change-points from partial observations effectively by exploiting potential correlations between sensors, and (ii) we formulate the sensor subset selection task as a Multi-Armed Bandit (MAB) problem and develop an effective adaptive sampling strategy using MAB algorithms. We offer theoretical analysis for the proposed online learning solution, and further validate its empirical performance via an extensive set of numerical studies together with a case study on real-world data sets.

Introduction

Unsupervised learning for sequential monitoring of complex systems (a.k.a., sequential change detection) is ubiquitous in a wide range of applications, such as quality control in manufacturing processes, logistic systems, transportation networks, electrical grids, the internet of things (IoT) systems, etc. Typically, to capture information about the underlying system state in real time, a massive array of variables are measured continuously and sequentially for prompt change detection and timely decision making. However, in practice, the acquisition of continuously measurements of such a large amount of variables may be infeasible due to resource constraints. For example, the number of available sensors may be less than the number of interested variables to be monitored due to expensive cost of the sensors; or at each sensing epoch only a limited number of sensors can be set in the "ON" mode for measurement due to limited battery

lifetime. Furthermore, when the number of variables is large and their measurement streams are generated in high velocity, real-time analysis may be significantly hindered due to the constraints of system memory, storage space, transmission bandwidth, computational power and processing speed. Consequently, in many applications, even if all variables can be measured, only partial observations of them can be transmitted back to the data fusion center for real-time analytics.

All the above constraints trigger the demand of new learning techniques to address the emerging challenge of Partially Observable Multi-sensor Sequential Change Detection (POMSCD), where only a subset of sensors can be observed at each epoch for change detection. Specifically, consider a system characterized by a set of p variables (for example, in a manufacturing system, each variable corresponds to a fabrication characteristic). Signals of these variables at each sensing epoch t are denoted as $\mathbf{X}(t) = [X_1(t), \ldots, X_p(t)]$, where each value $X_i(t), i = 1, \ldots, p$ is a scalar indicating the state of the ith system characteristic at the current epoch t. We are interested in detecting the distribution changes of the p variables in a sequential manner. However, due to the constraints of limited sensing resources, we can only choose to observe m out of p variables ($m < p$) at each sensing epoch. The goal of machine learning for POMSCD is to devise an online learning algorithm for sequential change detection which dynamically chooses a subset of variables to observe at each sensing epoch such that the detection power can be maximized without violating the sensing constraints.

Our key contributions of this paper are twofold: (i) we propose a sequential change detection scheme based on confidence region of the system state which is estimated via a Bayesian detection framework and can deal with missing observations. The proposed confidence region takes correlations of different variables into account and hence increases the detection power; (ii) As the detection power is closely related to the sensing resource allocation strategy, we formulate the learning problem as an online Combinatorial Multi-armed Bandit (CMAB) task (Hoi et al. 2018), and solve it using an effective adaptive sampling strategy. In particular, at each sensing epoch, the adaptive sampling strategy attempts to choose a subset of sensors/variables that are most informative to observe for change detection at the next epoch; (iii) Finally, we give theoretical analysis on the asymptotic detection power of the proposed adaptive sampling scheme.

Related Work

Our work is related to the family of sequential change-point detection works, which has been extensively studied in the literature of statistics, machine learning and beyond (Xie and Siegmund 2013; Tartakovsky, Nikiforov, and Basseville 2014; Chan 2017; Aminikhanghahi and Cook 2017). However, most existing studies assume all the variables with the target system can be fully observable, which are not directly applicable to POMSCD in the partially observable context. To the best of our knowledge, the only closely related works that attempt to address partial observations for multi-sensor change detection are Liu, Mei, and Shi (2015) and Xian, Wang, and Liu (2017). However, these two methods are traditional statistics-based approaches, and do not exploit variables' correlations. More critically, their detection powers are not satisfactory in some cases (which will be demonstrated in numerical studies) due to their heuristic adaptive sensor allocation strategies which are based on the rule of thumb and cannot give theoretical guarantee. By contrast, we propose the first online learning scheme for POMSCD with rigorous theoretical guarantee, which carefully exploits the correlation of variables and dynamically chooses the subset of informative sensors to maximize the detection power.

Our problem is also different from adaptive resource allocation problems such as ranking and selection (Nelson et al. 2001), statistical adaptive monitoring (Tartakovsky et al. 2006), multi-armed bandit (MAB)(Gai, Krishnamachari, and Jain 2010; Even-Dar, Mannor, and Mansour 2006; Bubeck, Wang, and Viswanathan 2013), active learning (Kapoor et al. 2007) and online feature selection (Wang et al. 2014), from the following perspectives: (i) In our problem, change patterns can be any mean shift and are unknown in advance; hence, unsupervised learning is more suitable. However, most of current change detection methods via active learning or feature selection are based on supervised learning, assuming abnormal patterns can be known beforehand for model training; (ii) We aim at a system-level change detection. When a change occurs, we aim to minimize the detection delay, while ranking and selection or traditional MAB studies aim at identifying which variable is changed (Zhuang, Wang, and Wang 2017), i.e., detection in the variable level. Consequently, they may not guarantee an effective system-level performance; (iii) We allow different variables to have correlations with each other, and such correlation information can help for efficient detection. However, most existing works across the above often assume the data streams are independent with each other.

Problem Formulation

Consider a system with p correlated variables, denote their signals at sensing epoch t as $\mathbf{X}(t) = [X_1(t), \ldots, X_p(t)]$. Initially, $\mathbf{X}(t)$ independently and identically (*i.i.d*) follows a joint distribution $p(\mathbf{X}(t)|\boldsymbol{\mu})$ with the mean vector $\boldsymbol{\mu}$. At some unknown time epoch τ, an unusual event (a change) occurs and affects an unknown subset of the p data streams in the sense that if variable i is affected, the mean of its local observations $X_i(t)$ changes from μ_i to μ_{ci} for epoch $t \geq \tau$.

The problem is to raise an alarm as quickly as possible after the change occurs. This mean (location) shift is a popular change pattern considered in many works, especially for statistical process control in manufacturing (Montgomery 2009). Without loss of generality, we set the before-change mean vector as $\boldsymbol{\mu} = \mathbf{0}$ when the system is in the normal condition, and set the post-change mean vector as $\boldsymbol{\mu}_c \neq \mathbf{0}$. Due to limited sensing resources, at each sensing epoch, we can only observe m out of p variables ($m \leq p$). By introducing the binary decision variable z_{it} for each variable $X_i(t)$ such that $z_{it} = 1$ if and only if $X_i(t)$ is observed at epoch t, the sensing constraint can be expressed as $\sum_{i=1}^{p} z_{it} = m, \forall t$. We denote by $Z(t)$ the vector of indices corresponding to the selected variables/sensors for observations in $\mathbf{X}(t)$.

A general detection scheme to decide whether a change occurs is defined as a scheme of stopping time T associated with a test statistic $\Lambda(t), \forall t$. Typically, one can define $T = \inf_t \{\Lambda(t) > h\}$ where h is a pre-defined constant threshold. The interpretation of T is that, when $T = n$, the scheme stops at epoch n and declares that a change has occurred in some variables somewhere in the first n time epochs. We want the scheme to stop as soon as possible after a change occurs (in the abnormal condition) but will continue monitoring without raising false alarms as long as possible if no change occurs (in the normal condition). In other words, we want $P(\Lambda(t) > h)$ as small as possible in the normal case and as large as possible in the abnormal case. The performance of a scheme T can be evaluated by two performance indicators: the Average Run Length (ARL) before a false alarm occurs in the normal case, i.e., $ARL_0 = E(T|\tau = \infty)$ and the Average Detection Delay (ADD) in the abnormal case, i.e., $ADD_\tau = E(T - \tau|T > \tau, \tau < \infty)$. In practice, the problem can be equivalently formulated as finding a change detection scheme T that minimizes the ADD performance subject to a constraint imposed on the ARL_0, i.e., ARL_0 must be greater than a predefined constant that essentially controls the global false alarm rate.

Our Method

Consider a general change detection scheme for POMSCD, at each epoch t, it first determines if the system is still in normal condition. If no, an alarm is triggered; otherwise, it determines the sampling strategy i.e., $Z(t + 1)$ to choose a subset of sensors to observe at the next epoch $t+1$ within the sensing constraint. In particular, we address the following two issues: (1) how to construct a change detection scheme for the partially observed data streams with consideration of their correlations; and (2) how to learn a good strategy to determine which m out of the p variables to be sampled at each epoch to increase the detection power of the scheme.

Online Weighted Bayesian Change Detection

Assume up to the current epoch n, no change alarm is triggered. Then based on $Z(t), t = 1, \ldots, n$, we can estimate the current system state, i.e., the mean vector $\boldsymbol{\mu}$ of the p variables. Here we adopt the Bayesian posterior estimator approach. Since samples at recent epochs are more likely to represent the current system state and can better identify the

potential system changes than the past epochs, we would like to stress more effort on the current epochs in the estimation. As such, we enforce time decayed weights $w_t^n, t = 1, \dots, n$ on the n samples, in the sense that $w_1^n < w_2^n < \dots < w_n^n$, and get the weighted posterior distribution of $\boldsymbol{\mu}$ is

$$\tilde{p}(\boldsymbol{\mu}|\mathbf{X}_{Z(1)}, \dots, \mathbf{X}_{Z(n)}) \propto p_0(\boldsymbol{\mu}) \prod_{t=1}^{n} p(\mathbf{X}_{Z(t)}|\boldsymbol{\mu})^{w_t^n}, \quad (1)$$

where $\mathbf{X}_{Z(t)}$ is the vector of observed variables of $\mathbf{X}(t)$, and $p_0(\boldsymbol{\mu})$ is the prior distribution. In this paper, we use the exponential decaying weight, i.e., $w_t^n = (1 - \lambda)^{n-t}$ with a small positive value $\lambda \in (0, 0.1]$. Then (1) can be written in an incremental way as

$$\tilde{p}(\boldsymbol{\mu}|\mathbf{X}_{Z(1)}, \dots, \mathbf{X}_{Z(n)}) = \quad (2)$$
$$\tilde{p}(\boldsymbol{\mu}|\mathbf{X}_{Z(1)}, \dots, \mathbf{X}_{Z(n-1)})^{1-\lambda} p(\mathbf{X}_{Z(n)}|\boldsymbol{\mu}).$$

Following many previous works(Xie and Siegmund 2013; Liu, Mei, and Shi 2015), in this paper we assume $\mathbf{X}(t)$ follows a Gaussian distribution with covariance matrix $\boldsymbol{\Sigma}$. Though there also exist some works that consider more general distributions, all the current works are constructed in a FULLY observed context, and cannot be applied in our case. Since our main contribution is change detection with ADAPTIVE SENSOR ALLOCATION for PARTIAL OBSERVED DATA STREAM, we focus on the most classical setting and would extend to more complex scenarios in future. In particular, by setting $p_0(\mathbf{u})$ to be non-informative, (2) also follows a Gaussian distribution with mean $\boldsymbol{\mu}_n$ and covariance matrix $\mathbf{V}_n$ which can be calculated incrementally as:

$$\mathbf{V}_n^{-1} = (1 - \lambda)\mathbf{V}_{n-1}^{-1} + \mathbf{E}_{Z(n)}' \boldsymbol{\Sigma}_{Z(n)}^{-1} \mathbf{E}_{Z(n)}, \quad (3)$$
$$\boldsymbol{\mu}_n = \mathbf{V}_n[(1 - \lambda)\boldsymbol{\mu}_{n-1} + \mathbf{E}_{Z(n)}' \boldsymbol{\Sigma}_{Z(n)}^{-1} \mathbf{X}_{Z(n)}].$$

Here $\boldsymbol{\Sigma}_{Z(t)}$ is the $Z(t)$ rows and columns of $\boldsymbol{\Sigma}$, and $\mathbf{E}_{Z(t)} \in \mathcal{R}^{m \times p}$. Its ith row is a unit vector who has 1 at the position of the ith element in $Z(t)$ and 0 otherwise.

Note that $\boldsymbol{\mu}_n$ of (3) is estimated by considering the correlation of different variables. For each variable i, its estimated μ_{ni} is not only based on the past observations of itself $X_i(t), t = 1, \dots, n$, but also on past observations of the other variables $X_j(t), j \neq i, t = 1, \dots, n$. In other words, even if variable i is not observed at epoch n, its mean μ_{ni} will still be updated according to its correlations with the other observed variables at epoch n.

Based on (3), we can get the $1 - \alpha$ confidence region of the current system mean vector $\boldsymbol{\mu}$ as $\mathcal{C}_\alpha = \{\mathbf{u}|(\mathbf{u} - \boldsymbol{\mu}_n)'\mathbf{V}_n^{-1}(\mathbf{u} - \boldsymbol{\mu}_n) \leq \chi_{p,1-\alpha}^2\}$, where $\chi_{p,1-\alpha}^2$ is the $1 - \alpha$ upper critical value of χ_p^2 distribution with p degree of freedom. Then if $\mathbf{0} \in \mathcal{C}_\alpha$, it means that we can not reject $\boldsymbol{\mu} \neq 0$ with $1 - \alpha$ confidence, and *vice versa*. As such, we can construct the test statistic as

$$\Lambda(n) = (\mathbf{0} - \boldsymbol{\mu}_n)'\mathbf{V}_n^{-1}(\mathbf{0} - \boldsymbol{\mu}_n) = \boldsymbol{\mu}_n'\mathbf{V}_n^{-1}\boldsymbol{\mu}_n. \quad (4)$$

Now we evaluate some theoretical properties of (4) in both normal and abnormal conditions.

Lemma 1 *When the system is still in the normal condition with $\boldsymbol{\mu} = \mathbf{0}$, as $\lambda \to 0$, $\Lambda(n)$ asymptotically follows a χ_p^2 distribution with p degree of freedom.*

Based on Lemma 1, we can set a detection threshold h for $\Lambda(n)$ according to a pre-specific confidence level α (false alarm rate), and define that if $\Lambda(n) > h$, the test statistic triggers an abnormal alarm.

Lemma 2 *When the system goes to an abnormal condition with $\boldsymbol{\mu} = \boldsymbol{\mu}_c$ since the first epoch with $\tau = 1$, as $\lambda \to 0$, $\Lambda(n)$ asymptotically follows a non-central chi-square distribution $\chi_p^2(c_n)$ with the non-centrality*

$$c_n = \boldsymbol{\mu}_c'\mathbf{V}_n^{-1}\boldsymbol{\mu}_c = \sum_{t=1}^{n} w_t^n \boldsymbol{\mu}_{cZ(t)}' \boldsymbol{\Sigma}_{Z(t)}^{-1} \boldsymbol{\mu}_{cZ(t)}.$$

Lemma 2 shows that the detection power of $\Lambda(n)$ is related to the non-centrality c_n. A higher c_n indicates a larger detection power. Furthermore, c_n depends on $\mathbf{V}_n$, which is further related to $Z(t), t = 1, \dots, n$. As such, it is desirable to construct an adaptive sampling strategy to maximize c_n.

Adaptive Sampling for Sensors Selection by Combinatorial Multi-armed Bandit

The problem of choosing a subset Z with $|Z| = m$ from p variables at each sensing epoch n is similar to MAB with multiple plays (Gai, Krishnamachari, and Jain 2010; Komiyama, Honda, and Nakagawa 2015; Xia et al. 2016; Zhou and Tomlin 2018). However, we note that in our problem c_n cannot be decomposed as a combination of m separate functions of $\mu_{ci}, i \in Z(t)$, i.e., not a linear function of the m selected variables. By contrast, different variables have cross-influence on each other via the precision matrix $\boldsymbol{\Phi}_{Z(t)} = \boldsymbol{\Sigma}_{Z(t)}^{-1}$. Furthermore, the cross-influence magnitude between two selected variables, say i and j, is further dependent on the other selected variables $z \in Z(t), z \neq i, j$, since $(\boldsymbol{\Phi}_{Z_1})_{ij} \neq (\boldsymbol{\Phi}_{Z_2})_{ij}$ if two sets $Z_1 \neq Z_2$. This means that we cannot simply select these m variables one by one.

Instead, we propose to re-formulate the problem as a Combinatorial Multi-armed Bandit (CMAB) task, where we treat a set Z as a *super arm*, with $M = \binom{p}{m}$ super arms in total, and our goal is to select the best super arm. Specifically, we define the set of all super arms as $\mathcal{Z} = \{Z_k, k = 1, \dots, M\}$. The reward of choosing Z_k at epoch t is

$$r_k(\boldsymbol{\mu}_c) = \boldsymbol{\mu}_{cZ_k}' \boldsymbol{\Phi}_{Z_k} \boldsymbol{\mu}_{cZ_k} = \sum_{i \in Z_k} \sum_{j \in Z_k} \phi_{ijZ_k} \mu_{ci} \mu_{cj}, \quad (5)$$

where ϕ_{ijZ_k} is the (i, j) component of $\boldsymbol{\Phi}_{Z_k}$. We are interested in designing a sampling strategy for this CMAB problem that performs well with respect to *time decayed expected regret*, which is defined as

$$R_n^\pi(\boldsymbol{\mu}_c) = \sum_{t=1}^{n} w_t^n r^\star - E_\pi(\sum_{t=1}^{n} w_t^n r_{\pi(t)}(\boldsymbol{\mu}_c)), \quad (6)$$

where $r^\star = \boldsymbol{\mu}_{cZ^\star}' \boldsymbol{\Phi}_{Z^\star} \boldsymbol{\mu}_{cZ^\star}$ is the reward of the optimal super arms. Here $Z^\star$ denote optimal arms, which are not necessarily unique. In particular, denote the true changed variable set of the p variables as Z^a, with the cardinality $|Z^a| = a$. If $a \leq m$, the optimal super arms $Z^\star$ are any set that includes Z^a, i.e., $Z^a \subseteq Z^\star$. If $m < a$, the optimal super arm is a unique subset of Z^a satisfying

$Z^\star = \arg\max_{Z \subset Z^a, |Z|=m} \boldsymbol{\mu}'_{cZ} \boldsymbol{\Phi}_Z \boldsymbol{\mu}_{cZ}$. In other words, the optimal sampling strategy is to select all the changed arms $i \in Z^a$ (if $a \leq m$) or select the best subset of the changed arms (if $a > m$).

Though CMAB has attracted increasing attention recently, most of existing works focus on linear reward functions(Gai, Krishnamachari, and Jain 2010; Durand and Gagné 2014). However, as mentioned earlier, our reward function $r_k(\boldsymbol{\mu}_c)$ is nonlinear with the individual arms $\mu_{ci}, i \in Z_k$. So far to our best knowledge, the only related work for CMAB with non-linear reward functions is Chen et al. (2016). However, this work has additional assumptions that the reward functions of super arms should have the monotonicity property: *the expected reward of playing any super arm $Z \in \mathcal{Z}$ is monotonically nondecreasing with respect to their expectation vector, i.e., if for all $i = 1, \ldots, p$, $\mu_{ci} \leq \mu'_{ci}$, we have $r_Z(\boldsymbol{\mu}_c) \leq r_Z(\boldsymbol{\mu}'_c), \forall Z \in \mathcal{Z}$.* Clearly, this cannot be satisfied in our case. Consequently, direct applying these sampling strategies in the literature to our case would lose their regret bound and lead to poor detection power. As such, in this paper we develop a new CMAB strategy tailored for our reward function, based on the UCB algorithm. In particular, for each variable i at epoch n, the $1 - 2\text{InvNorm}(\sqrt{\gamma_n})$ confidence interval of its mean μ_i is $\left[\mu_{ni} - \sqrt{\gamma_n v_{ni}}, \mu_{ni} + \sqrt{\gamma_n v_{ni}}\right]$, where v_{ni} is the i diagonal item of $\mathbf{V}_n$ representing the variance of the posterior distribution of μ_i. Then the upper confidence bound of the estimated reward of choosing Z_k at epoch t based on $\boldsymbol{\mu}_n$ is

$$\sum_{i \in Z_k} \sum_{j \in Z_k} \phi_{ijZ_k}\left(\mu_{ni} + \text{sgn}(\phi_{ijZ_k}\mu_{ni}\mu_{nj})\text{sgn}(\mu_{ni})\sqrt{\gamma_n v_{ni}}\right)$$
$$\tag{7}$$
$$\times \left(\mu_{nj} + \text{sgn}(\phi_{ijZ_k}\mu_{ni}\mu_{nj})\text{sgn}(\mu_{nj})\sqrt{\gamma_n v_{nj}}\right)$$
$$= \sum_{i \in Z_k} \sum_{j \in Z_k} \phi_{ijZ_k}\mu_{ni}\mu_{nj} + \sqrt{\gamma_n \phi_{ijZ_k}^2 \mu_{ni}^2 v_{nj}}$$
$$+ \sqrt{\gamma_n \phi_{ijZ_k}^2 \mu_{nj}^2 v_{in}} + \gamma_n\sqrt{\phi_{ijZ_k}^2 v_{ni}v_{nj}},$$

where $\text{sgn}(x) = 1, \forall x \geq 0$ and $\text{sgn}(x) = -1, \forall x < 0$. The first part of (7) emphasizes on exploitation of the best super arm so far we estimate, while the last three parts of (7) emphasize on exploration of other potential super arms by considering the estimation uncertainty. The term γ_n balances exploitation and exploration. The bigger it gets, the more it favours arms with high v_{ni} (exploration). If $\gamma_n = 0$, the algorithm is greedy. In our algorithm, we set γ_n as $\exp(\gamma_n) = 2(1 - (1-\lambda)^n)/\lambda$. Then we can select the supper arm Z_k that maximizes (7) as the variable set to be sampled for epoch $n + 1$. The detailed adaptive sampling strategy is shown in Algorithm 1.

Theorem 1 *By setting $\gamma_n = 2\log(\frac{1-(1-\lambda)^n}{\lambda^2})$, the $R_n^\pi(\boldsymbol{\mu}_c)$ of Algorithm 1 is at most*

$$\left[2\left(\max_{Z_l} \frac{4\tilde{\boldsymbol{\mu}}'_{Z_l}\tilde{\boldsymbol{\Phi}}_{Z_l}\mathbf{I} + 2||\tilde{\boldsymbol{\Phi}}_{Z_l}||_1}{\Delta_{Z_l}}\right)^q \log(\frac{1-(1-\lambda)^n}{\lambda^2}) + 2m(1-(1-\lambda)^n)\right]p\Delta_{max},$$

Algorithm 1: CMAB for adaptively sampling Z_t

Input: Data streams $\mathbf{X}_n, n = 1, \ldots,$
initialize $\mathbf{v}_m, m = 1, \ldots, M$
for $n = 1, \ldots, \lceil p/m \rceil$ **do**
   Observe any m variables that have not been observed so far.
   Update $\boldsymbol{\mu}(n)$ and $\mathbf{V}_n$ according to (3).
for $n = \lceil p/m \rceil + 1, \ldots$ **do**
   Select $Z \in \mathcal{Z}$ for $Z(n+1)$ by maximizing

$$\sum_{i \in Z} \sum_{j \in Z} \phi_{ijZ}\mu_{ni}\mu_{nj} + \sqrt{\gamma_n \phi_{ijZ}^2 \mu_{ni}^2 v_{nj}} \quad (8)$$
$$+ \sqrt{\gamma_n \phi_{ijZ}^2 \mu_{nj}^2 v_{ni}} + \gamma_n\sqrt{\phi_{ijZ}^2 v_{ni}v_{nj}}.$$

---

where $\tilde{\boldsymbol{\mu}}_{Z(t)} \in \mathcal{R}^{m \times 1}$ with elements $\{|\mu_{cj}|\}_{j \in Z(t)}$; $\tilde{\boldsymbol{\Phi}}_{Z(t)} \in \mathcal{R}^{m \times m}$ with elements $\{|\phi_{ij}|\}_{i,j \in Z(t)}$; $\Delta_{Z_l} = \boldsymbol{\mu}'_{cZ^\star}\boldsymbol{\Phi}_{Z^\star}\boldsymbol{\mu}_{cZ^\star} - \boldsymbol{\mu}'_{cZ_l}\boldsymbol{\Phi}_{Z_l}\boldsymbol{\mu}_{cZ_l}$; $q = 1$ if $\max_{Z_l} \frac{4\tilde{\boldsymbol{\mu}}'_{Z_l}\tilde{\boldsymbol{\Phi}}_{Z_l}\mathbf{I} + 2||\tilde{\boldsymbol{\Phi}}_{Z_l}||_1}{\Delta_{Z_l}} \leq 1$ *while $q = 2$ otherwise.*

Proof 1 *To derive and bound the decayed expected regret of Algorithm 1, we can analyze the expected number of times that each non-optimal super arm is played, and sum this expectation over all non-optimal super arms. In particular, we have*

$$R_n^\pi = E_\pi[\sum_{t=1}^n w_t^n r^\star - \sum_{t=1}^n w_t^n r_{\pi(t)}(\boldsymbol{\mu}_c)] \leq \Delta_{max} \sum_{l \in \mathcal{Z}} E[T_l(n)],$$

where $T_l(n)$ means the weighted number of epochs that Z_l is chosen from epoch 1 up to epoch n.

We introduce $\{\tilde{T}_j(n)\}, j = 1, \ldots, p, n = 1, \ldots$ as a counter after the initialization period. It is updated in the following way: At each time epoch after the initialization period, one of the two cases must happen: (1) an optimal super arm is played; (2) a non-optimal super arm is played. In the first case, set $\tilde{T}_j(n) = \tilde{T}_j(n-1), \forall j = 1, \ldots, p$, and $\tilde{T}_j(t) = \tilde{T}_j(t) \cdot (1 - \lambda), \forall j = 1, \ldots, p, t = 1, \ldots, n-1$. When a non-optimal super arm Z_k is picked at time n, besides updating $T_j(n)$ by the same way as the first case, we choose $j' \in Z_k$ such that $j' = \arg\min_{j \in Z_k} T_j(n-1)$. If there are multiple such arms, we arbitrarily pick one, say j', and increase $T_{j'}(n)$ by 1. In this way, each time a non-optimal arm is picked, exactly one element in $\tilde{T}_j(n)$ is incremented by 1. This implies that the total "weighted" number we have played the non-optimal super arms up to epoch n is equal to the sum of all counters in $\{\tilde{T}_j(n)\}_{1:p}$. Therefore, we have

$$\sum_{l \in \mathcal{Z}} E[T_l(n)] = \sum_{j=1}^p E[\tilde{T}_j(n)].$$

In the proof, for explanation clarity, we abuse notation and denote $\boldsymbol{\mu}_n$ as $\hat{\boldsymbol{\mu}}_n$. Without loss of generality, we further assume $\text{diag}(\boldsymbol{\Sigma}) = \mathbf{1}$ by assuming all the data have been normalized. The extension to other $\boldsymbol{\Sigma}$ is straight-forward. Denote $\tilde{I}_j(n)$ as the indicator function which is equal to 1 if

$\tilde{T}_j(n)$ *is added by one at time* n. *Let* d *be an arbitrary positive integer. Then*

$$\tilde{T}_j(n) = \sum_{t=1}^{n} w_t^n \{\tilde{I}_j(t)\} \le d + \sum_{t=2}^{n} w_t^n \{\tilde{I}_j(t), \tilde{T}_j(t-1) \ge d\}. \quad (9)$$

When $\tilde{I}_j(t) = 1$, *a non-optimal super arm* $Z(t)$ *is picked from which arm* $j = \arg\min_{j \in Z(t)} \tilde{T}_j(t-1)$ *is selected to update. Then we have*

$$\tilde{T}_j(n) \le d + \sum_{t=2}^{n} w_t^n \{ \sum_{i \in Z^\star} \sum_{j \in Z^\star} \phi_{ij}^\star \hat{\mu}_{ti} \hat{\mu}_{tj} + \sqrt{\gamma_t \phi_{ij}^{\star 2} \hat{\mu}_{ti}^2 v_{tj}}$$
$$+ \sqrt{\gamma_t \phi_{ji}^{\star 2} \hat{\mu}_{tj}^2 v_{ti}} + \gamma_t \sqrt{\phi_{ij}^{\star 2} v_{ti} v_{tj}}$$
$$\le \sum_{i \in Z(t)} \sum_{j \in Z(t)} \phi_{ijZ(t)} \hat{\mu}_{ti} \hat{\mu}_{tj} + \sqrt{\gamma_t \phi_{ijZ(t)}^2 \hat{\mu}_{ti}^2 v_{tj}}$$
$$+ \sqrt{\gamma_t \phi_{jiZ(t)}^2 \hat{\mu}_{tj}^2 v_{ti}} + \gamma_t \sqrt{\phi_{ijZ(t)}^2 v_{ti} v_{tj}}, \tilde{T}_j(t-1) \ge d \}.$$

Note that $d \le \tilde{T}_j(t-1) = \min_{j \in Z(t)} \tilde{T}_j(t-1)$, *this indicates* $T_j(t) \ge d, \forall j \in Z(t)$. *The second part of* (9) *indicates that at least one of the following must be true:*

$$\sum_{i \in Z^\star} \sum_{j \in Z^\star} \phi_{ij}^\star \hat{\mu}_{ti} \hat{\mu}_{tj} \le \sum_{i \in Z^\star} \sum_{j \in Z^\star} \phi_{ij}^\star \mu_{ci} \mu_{cj} - \sqrt{\gamma_t \phi_{ij}^{\star 2} \hat{\mu}_{ti}^2 v_{tj}}$$
$$\quad (10)$$
$$- \sqrt{\gamma_t \phi_{ji}^{\star 2} \hat{\mu}_{tj}^2 v_{ti}} - \gamma_t \sqrt{\phi_{ij}^{\star 2} v_{ti} v_{tj}};$$

$$\sum_{i \in Z(t)} \sum_{j \in Z(t)} \phi_{ijZ(t)} \hat{\mu}_{ti} \hat{\mu}_{tj} \ge \sum_{i \in Z(t)} \sum_{j \in Z(t)} \phi_{ijZ(t)} \mu_{ci} \mu_{cj} \quad (11)$$
$$+ \sqrt{\gamma_t \phi_{ijZ(t)}^2 \hat{\mu}_{ti}^2 v_{tj}} + \sqrt{\gamma_t \phi_{jiZ(t)}^2 \hat{\mu}_{tj}^2 v_{ti}} + \gamma_t \sqrt{\phi_{ijZ(t)}^2 v_{ti} v_{tj}};$$

$$\sum_{i \in Z^\star} \sum_{j \in Z^\star} \phi_{ij}^\star \mu_{ci} \mu_{cj} \le \sum_{i \in Z(t)} \sum_{j \in Z(t)} \phi_{ijZ(t)} \mu_{ci} \mu_{cj} \quad (12)$$
$$+ 2\sqrt{\gamma_t \phi_{ijZ(t)}^2 \hat{\mu}_{ti}^2 v_{tj}} + 2\sqrt{\gamma_t \phi_{jiZ(t)}^2 \hat{\mu}_{tj}^2 v_{ti}} + 2\gamma_t \sqrt{\phi_{ijZ(t)}^2 v_{ti} v_{tj}}.$$

Now we first find the probability upper bound for (10). *In particular,*

$$P[(10)] \le \sum_{i \in Z^\star} P[\mu_{ci} - \hat{\mu}_{ti} \ge \sqrt{\gamma_t v_{ti}}] \le \sum_{i \in Z^\star} \exp(-\frac{\gamma_t v_{ti}}{2 v_{ti}}),$$

by setting $\gamma_n = 2\log(\frac{1-(1-\lambda)^n}{\lambda^2})$, *we have* $P[(10)] \le \frac{m\lambda^2}{1-(1-\lambda)^t}$.

Similarly, we can also get $P[(11)] \le \frac{m\lambda^2}{1-(1-\lambda)^t}$.
Now we derive the upper bound of (12).

$$P[(12)] = P \left[\Delta_{Z(t)} \le \sum_{i \in Z(t)} \sum_{j \in Z(t)} 2\sqrt{\gamma_t \phi_{ijZ(t)}^2 \hat{\mu}_{ti}^2 v_{tj}} \right.$$
$$+ 2\sqrt{\gamma_t \phi_{jiZ(t)}^2 \hat{\mu}_{tj}^2 v_{ti}} + 2\gamma_t \sqrt{\phi_{ijZ(t)}^2 v_{ti} v_{tj}} \right]$$
$$\le P \left[\Delta_{Z(t)} \le 4\tilde{\boldsymbol{\mu}}_{Z(t)}' \tilde{\boldsymbol{\Phi}}_{Z(t)} \mathbf{I} \sqrt{\frac{\gamma_t}{d}} + 2||\tilde{\boldsymbol{\Phi}}_{Z(t)}||_1 \frac{\gamma_t}{d} \right].$$

If $d \ge \gamma_t$, *we have*

$$p[(12)] \le P \left[\Delta_{Z(t)} \le (4\tilde{\boldsymbol{\mu}}_{Z(t)}' \tilde{\boldsymbol{\Phi}}_{Z(t)} \mathbf{I} + 2||\tilde{\boldsymbol{\Phi}}_{Z(t)}||_1) \sqrt{\frac{\gamma_t}{d}} \right].$$

This means that as long as we set $d \ge \left(\max_{Z_l} \frac{4\tilde{\boldsymbol{\mu}}_{Z_l}' \tilde{\boldsymbol{\Phi}}_{Z_l} \mathbf{I} + 2||\tilde{\boldsymbol{\Phi}}_{Z_l}||_1}{\Delta_{Z_l}} \right)^2 \gamma_t$ *the condition of* (12) *is false. If* $d < \gamma_t$, *we have*

$$p[(12)] \le P \left[\Delta_{Z(t)} \le (4\tilde{\boldsymbol{\mu}}_{Z(t)}' \tilde{\boldsymbol{\Phi}}_{Z(t)} \mathbf{I} + 2||\tilde{\boldsymbol{\Phi}}_{Z(t)}||_1) \frac{\gamma_t}{d} \right]$$

This means that as long as we set $d \ge \left(\max_{Z_l} \frac{4\tilde{\boldsymbol{\mu}}_{Z_l}' \tilde{\boldsymbol{\Phi}}_{Z_l} \mathbf{I} + 2||\tilde{\boldsymbol{\Phi}}_{Z_l}||_1}{\Delta_{Z_l}} \right) \gamma_t$ *the condition of* (12) *is false.*

$$E[\tilde{T}_j(n)] \le 2 \left(\max_{Z_l} \frac{4\tilde{\boldsymbol{\mu}}_{Z_l}' \tilde{\boldsymbol{\Phi}}_{Z_l} \mathbf{I} + 2||\tilde{\boldsymbol{\Phi}}_{Z_l}||_1}{\Delta_{Z_l}} \right)^q \log(\frac{1-(1-\lambda)^n}{\lambda^2})$$
$$+ 2m \sum_{t=1}^{n} \frac{\lambda^2}{1-(1-\lambda)^t} (1-\lambda)^{n-t}$$
$$\le 2 \left(\max_{Z_l} \frac{4\tilde{\boldsymbol{\mu}}_{Z_l}' \tilde{\boldsymbol{\Phi}}_{Z_l} \mathbf{I} + 2||\tilde{\boldsymbol{\Phi}}_{Z_l}||_1}{\Delta_{Z_l}} \right)^q \log(\frac{1-(1-\lambda)^n}{\lambda^2})$$
$$+ 2m(1-(1-\lambda)^n),$$

where $q = 1$ *if* $\max_{Z_l} \frac{4\tilde{\boldsymbol{\mu}}_{Z_l}' \tilde{\boldsymbol{\Phi}}_{Z_l} \mathbf{I} + 2||\tilde{\boldsymbol{\Phi}}_{Z_l}||_1}{\Delta_{Z_l}} \le 1$ *and* $q = 2$ *otherwise. Consequently, we can conclude Theorem 1.*

Corollary 1 *As* $n \to \infty$, $1 - (1-\lambda)^n \to 1$, *which is fixed. Then we have* $R_n^\pi(\boldsymbol{\mu}_c)$ *is asymptotically bounded by*

$$R_\infty = \left[4 \left(\max_{Z_l} \frac{4\tilde{\boldsymbol{\mu}}_{Z_l}' \tilde{\boldsymbol{\Phi}}_{Z_l} \mathbf{I} + 2||\tilde{\boldsymbol{\Phi}}_{Z_l}||_1}{\Delta_{Z_l}} \right)^q \log(\frac{1}{\lambda}) + 2m \right] p\Delta_{max}.$$

With Corollary 1, we can bound the asymptotic expected detection power of $\Lambda(n)$.

Corollary 2 *When* $\lambda \to 0$ *and* $\tau = 1$, *the asymptotic expected detection power of* $\Lambda(n)$ *as* $n \to \infty$ *based on the adaptive sampling strategy in Algorithm 1 is bigger than*

$$1 - Q_{\frac{p}{2}}(\sqrt{c^\star - R_\infty}, \sqrt{h}),$$

where $c^\star = \boldsymbol{\mu}_{cZ^\star}' \boldsymbol{\Phi}_{Z^\star} \boldsymbol{\mu}_{cZ^\star}$, *and* $Q_{p/2}$ *is the Marcum Q-function.*

The whole proposed change detection scheme is as follows. For each new online sample $\mathbf{X}(n)$, we first observe its values $\mathbf{X}_{Z(n)}$ and get $\Lambda(n)$ according to (3) and (4). If $\Lambda(n) > h$, the scheme triggers an abnormal alarm and the system stops for diagnosis. Otherwise, the scheme decides $Z(n+1)$ by Algorithm 1.

Remark 1 *When* m *or* p *is big, searching all the* $M = \binom{p}{m}$ *super arms might be time consuming, and the algorithm might be impractical for very large-scale problems with limited data processing resource. Therefore a more time-efficient algorithm is desirable. Recall that it is* $\boldsymbol{\Phi}_{Z(t)}$ *that makes* c_n *a nonlinear function of the selected variables. If we force* $\boldsymbol{\Phi}_{Z(t)}$ *to be a diagonal matrix, we can remove all the cross-influence and make* c_n *as the simplified* $\tilde{c}_n = \sum_{t=1}^{n} \sum_{i \in Z(t)} \mu_{ci}^2$, *which is a linear (and monotone) function of the square of each arm's reward,* μ_{ci}^2. *In this case, sampling* Z *based on* (8) *in Algorithm 1 is degenerated to the following strategy: calculating*

$$r_i = |\mu_{ni}| + \sqrt{\gamma_n v_{ni}}, \quad (13)$$

ranking r_i from the largest to smallest as $r_{(1)} \geq r_{(2)} \geq \ldots \geq r_{(p)}$, and selecting the variables with $r_{(1)}, \ldots, r_{(m)}$ as $Z(n+1)$. The selection by (13) is similar to the CMAB strategy in Chen et al. (2016).

In the following experiments, we denote the proposed detection scheme for POMSCD using the CMAB algorithm in Algorithm 1 with strategy (8) as "CMAB", and the detection scheme for POMSCD using the simplified CMAB strategy in (13) as "CMAB(s)" for short.

Numerical Studies

We conduct extensive experiments on both synthetic and real-world data sets, to evaluate the performance of the proposed POMSCD detection schemes with the two CMAB strategies, i.e., "CMAB" and "CMAB(s)". We compare our schemes with the following existing baselines:

- **TRAS**: the top-r adaptive sampling detection algorithm in Liu, Mei, and Shi (2015);
- **NAS**: the nonparametric anti-rank adaptive sampling algorithm in Xian, Wang, and Liu (2017);

In addition, we also include two additional variants of our detection scheme as follows:

- **RAND**: a variant of the proposed POMSCD scheme by replacing the adaptive sampling strategies (CMAB or CMAB(s)) with a **random** strategy, i.e., $Z(n+1)$ is drawn by randomly sampling m from a total of p variables;
- **ORACLE**: the proposed detection scheme for POMSCD but assuming all the p variables are *fully observable* at each epoch. Clearly, this is just an oracle scheme and used as a performance upper bound of our detection scheme.

Synthetic Data Experiments

Our experiments considers two settings, $p = 10$ and $p = 100$ respectively. For both settings, we set $(\boldsymbol{\Sigma})_{ij} = 0.5, \forall i \neq j$, $m = 5$ and $\lambda = 0.1$, and consider the following two change patterns: (i) The first q variables have the same mean shift magnitude with the same sign, i.e., $\boldsymbol{\mu}_c = [\underbrace{1, \ldots, 1}_{q}, 0, \ldots, 0] \times \delta$; (ii) The first q variables have the same mean shift magnitude with opposite signs, i.e., $\boldsymbol{\mu}_c = [\underbrace{1, -1, 1, \ldots, 1, -1}_{q}, 0, \ldots, 0] \times \delta$; where δ is the shift magnitude. We set $q = 4$ for $p = 10$ and $q = 16$ for $p = 100$. For each algorithm, we set its h such that its $ARL_0 = 200$, and we evaluate its detection performance in terms of ADD_{50} with $\tau = 50$. All the ADDs in subsequent simulations are calculated based on 10000 simulation replications. For TRAS, we set its parameter $r = m$, $\mu_{min} = 0.25$ and $\Delta = 0.03$ for both settings according to the recommendation of Liu, Mei, and Shi (2015). For NAS, we set its parameter $k = 0.5$, and $\Delta = 0.04$ for $p = 10$ and $\Delta = 0.1$ for $p = 100$ following the algorithm of Xian, Wang, and Liu (2017). Their performances for the two change patterns are shown in Tables 1-2.

It is clear that for both change patterns in both settings, except for ORACLE that is infeasible in practice, CMAB

has the smallest ADD, followed by CMAB(s). Compared with these two schemes, RAND has weaker detection power, which validates the efficacy of the proposed adaptive sampling algorithm. However, when p is small (e.g., $p = 10$), RAND still outperforms TRAS and NAS, indicating the superiority of the proposed online change detection scheme for POMSCD. Furthermore, for the same shift magnitude, the proposed detection schemes (CMAB, CMAB(s), RAND and ORACLE) generally have better detection performance on pattern (ii) than pattern (i). This is reasonable since the detection power is largely determined by the Mahalanobis distance of the shifted mean vector from the normal one, i.e., $\Delta = \boldsymbol{\mu}_c' \boldsymbol{\Sigma}^{-1} \boldsymbol{\mu}_c$. Given $\boldsymbol{\Sigma}$ and the change patterns in our study, we have $\Delta_{(ii)} > \Delta_{(i)}$. However, as TRAS does not consider the correlations of variables in the detection scheme, it is not expected to have a better detection result for pattern (ii) than pattern (i). For NAS, it has the weakest detection power for pattern (i), since its test statistic is based on the ranks of different variables. When different variables shift in the same direction, their ranks will not change very much, and consequently NAS would have unsatisfactory performance.

To further examine the influence of m on the detection performance, Table 3 shows the ADDs of different methods on change pattern (i) with $\delta = 0.5$ using different m, in the setting $p = 10$. Clearly, as m increases, all the methods have smaller ADDs, and their relative performances are similar to those in Tables 1-2, where CMAB has the best performance.

Case Study on Real-world Data

We use a real data set from a semiconductor manufacturing process to demonstrate the application of our methodology. The data set is publicly available in UCI Machine Learning Repository (http://archive.ics.uci.edu/ml/datasets/SECOM). It contains in total 1,567 wafer samples from a semiconductor manufacturing process. Among them, 1,463 samples are classified as conforming ones (normal samples), while the remaining 104 samples are classified as nonconforming ones (abnormal samples). After preprocessing, each sample consists of measurements of 31 variables in each wafer production. Figure 1 shows the correlations of the 31 variables calculated based on all normal samples. We can see that different variables have strong cross-correlations with each other. This indicates considering variable correlations in the anomaly detection scheme is essential.

We construct the online change detection context as follows: for every simulation replication, at each sensing epoch $t < \tau$, we draw a sample $\mathbf{X}(t)$ from the 1,463 normal samples randomly and sequentially with replacement. At each sensing epoch $\tau \leq t$, we draw a sample $\mathbf{X}(t)$ from the 103 abnormal samples randomly and sequentially with replacement. We set $\tau = 50$. Furthermore, though the data set has a total of 31 variables for each sample, in our experiment, we assume only m out of 31 variables can be observed for each $\mathbf{X}(t)$. Similar to synthetic data experiments, we apply the change detection schemes to detect the abnormal wafer samples from the streams. Their calculated ADDs based on different m are shown in Table 4. As expected, CMAB achieves the best performance and both CMAB and CMAB(s) signif-

Table 1: Average Detection Delays (ADDs) of different methods for change pattern (i).

			$p = 10$						$p = 100$			
δ	NAS	TRAS	RAND	CMAB(s)	CMAB	ORACLE	NAS	TRAS	RAND	CMAB(s)	CMAB	ORACLE
0	200	200	200	200	200	200	200	200	200	200	200	200
0.25	165	81.2	81.6	34.3	25.4	13.9	198	151	196	137	114	41.3
0.5	157	41.1	24.4	9.15	6.94	4.60	197	91.9	195	61.3	48.5	6.31
0.75	154	28.3	8.68	4.83	3.87	3.04	196	69.5	193	23.5	16.5	3.35
1	152	21.3	4.46	3.50	2.95	2.24	196	57.5	190	13.1	8.91	2.44
1.25	140	17.3	3.96	2.56	2.01	1.85	194	52.5	187	8.42	5.43	2.08
1.5	135	14.3	3.14	2.14	1.87	1.65	195	45.6	152	5.53	4.90	1.82
1.75	132	12.6	2.90	1.75	1.52	1.48	192	42.9	32.7	4.43	3.86	1.65
2	129	11.2	1.76	1.61	1.44	1.38	194	39.0	17.2	3.99	3.45	1.50

Table 2: Average Detection Delays (ADDs) of different methods for change pattern (ii).

			$p = 10$						$p = 100$			
δ	NAS	TRAS	RAND	CMAB(s)	CMAB	ORACLE	NAS	TRAS	RAND	CMAB(s)	CMAB	ORACLE
0.25	109	119	55.4	24.1	19.4	10.6	168	178	198	136	112	30.3
0.5	50.5	81.1	16.8	6.97	5.03	3.64	87.3	90.7	196	63.5	49.3	5.03
0.75	28.5	40.6	7.50	4.08	3.12	2.50	57.2	68.9	195	18.4	11.2	3.23
1	19.5	28.1	4.60	3.13	2.21	2.01	36.2	57.0	193	9.12	5.32	2.42
1.25	15.1	21.2	3.12	2.34	2.01	1.84	29.5	49.7	190	5.78	4.14	1.93
1.5	13.3	17.3	2.62	2.07	1.98	1.61	20.1	45.1	101	4.27	3.64	1.69
1.75	11.7	14.1	2.22	1.84	1.72	1.58	19.6	42.3	24.8	3.98	2.95	1.65
2	11.3	12.6	1.92	1.71	1.64	1.57	17.5	38.8	14.9	3.11	2.49	1.48
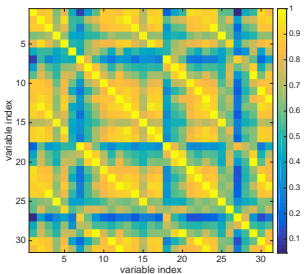
Figure 1: Correlations of the 31 variables of normal samples in the real-world data application.

Table 3: ADDs of different methods for pattern (i) with different m for $\delta = 0.5$ in $p = 10$.

m	NAS	TRAS	RAND	CMAB(s)	CMAB
3	185	78.2	31.6.	20.3	15.2
5	157	41.1	24.4	9.15	6.94
7	78.2	36.5	14.1	7.25	6.12
9	42.1	28.7	10.0	6.24	5.21
10	21.4	19.1	7.24	5.15	4.85

Table 4: ADDs of different methods on the real data set.

m	NAS	TRAS	RAND	CMAB(s)	CMAB
3	158	91.7	62.4	59.1	37.4
5	132	71.5	48.3	44.3	28.0
10	121	55.7	41.3	39.1	24.2
20	114	48.2	35.9	29.1	19.9
30	106	40.1	30.4	24.1	16.4

icantly outperform the baselines. This further confirms the superiority of the proposed methodology.

Conclusions

This paper presented a novel unsupervised online learning scheme for Partially Observable Multi-senor Sequential Change-point Detection (POMSCD) in the context of limited sensing resources available for monitoring changes of a system with multivariate streaming data. We tackled two open challenges of POMSCD: (i) how to construct an effective online change-point detection scheme from multi-sensor data streams with partial observations; and (ii) how to adaptively allocate sensing resources to collect the most informative observation in order to maximize the detection power.

Acknowledgements

References

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

Bubeck, S.; Wang, T.; and Viswanathan, N. 2013. Multiple identifications in multi-armed bandits. In *International Conference on Machine Learning*, 258–265.

Chan, H. P. 2017. Optimal sequential detection in multi-stream data. *The Annals of Statistics* 45(6):2736–2763.

Chen, W.; Wang, Y.; Yuan, Y.; and Wang, Q. 2016. Combinatorial multi-armed bandit and its extension to probabilistically triggered arms. *The Journal of Machine Learning Research* 17(1):1746–1778.

Durand, A., and Gagné, C. 2014. Thompson sampling for combinatorial bandits and its application to online feature selection. In *Workshops at the Twenty-Eighth AAAI Conference on Artificial Intelligence*.

Even-Dar, E.; Mannor, S.; and Mansour, Y. 2006. Action elimination and stopping conditions for the multi-armed bandit and reinforcement learning problems. *Journal of machine learning research* 7(Jun):1079–1105.

Gai, Y.; Krishnamachari, B.; and Jain, R. 2010. Learning multiuser channel allocations in cognitive radio networks: A combinatorial multi-armed bandit formulation. In *New Frontiers in Dynamic Spectrum, 2010 IEEE Symposium on*, 1–9. IEEE.

Hoi, S. C.; Sahoo, D.; Lu, J.; and Zhao, P. 2018. Online learning: A comprehensive survey. *CoRR* abs/1802.02871.

Kapoor, A.; Grauman, K.; Urtasun, R.; and Darrell, T. 2007. Active learning with gaussian processes for object categorization. In *Computer Vision, 2007. ICCV 2007. IEEE 11th International Conference on*, 1–8. IEEE.

Komiyama, J.; Honda, J.; and Nakagawa, H. 2015. Optimal regret analysis of thompson sampling in stochastic multi-armed bandit problem with multiple plays. In *International Conference on Machine Learning*, 1152–1161.

Liu, K.; Mei, Y.; and Shi, J. 2015. An adaptive sampling strategy for online high-dimensional process monitoring. *Technometrics* 57(3):305–319.

Montgomery, D. C. 2009. *Introduction to statistical quality control*. John Wiley & Sons (New York).

Nelson, B. L.; Swann, J.; Goldsman, D.; and Song, W. 2001. Simple procedures for selecting the best simulated system when the number of alternatives is large. *Operations Research* 49(6):950–963.

Tartakovsky, A. G.; Rozovskii, B. L.; Blazek, R. B.; and Kim, H. 2006. A novel approach to detection of intrusions in computer networks via adaptive sequential and batch-sequential change-point detection methods. *IEEE Transactions on Signal Processing* 54(9):3372–3382.

Tartakovsky, A.; Nikiforov, I.; and Basseville, M. 2014. *Sequential analysis: Hypothesis testing and changepoint detection*. Chapman and Hall/CRC.

Wang, J.; Zhao, P.; Hoi, S. C.; and Jin, R. 2014. Online feature selection and its applications. *IEEE Transactions on Knowledge and Data Engineering* 26(3):698–710.

Xia, Y.; Qin, T.; Ma, W.; Yu, N.; and Liu, T.-Y. 2016. Budgeted multi-armed bandits with multiple plays. In *IJCAI*, 2210–2216.

Xian, X.; Wang, A.; and Liu, K. 2017. A nonparametric adaptive sampling strategy for online monitoring of big data streams. *Technometrics* 1–12.

Xie, Y., and Siegmund, D. 2013. Sequential multi-sensor change-point detection. *The Annals of Statistics* 41(2):670–692.

Zhou, D. P., and Tomlin, C. J. 2018. Budget-constrained multi-armed bandits with multiple plays. In *AAAI*.

Zhuang, H.; Wang, C.; and Wang, Y. 2017. Identifying outlier arms in multi-armed bandit. In *Advances in Neural Information Processing Systems*, 5210–5219.