Online Anomaly Detection Under Markov Statistics With Controllable Type-I Error

Huseyin Ozkan, Fatih Ozkan, and Suleyman S. Kozat, *Senior Member, IEEE*

***Abstract*—We study anomaly detection for fast streaming temporal data with real time Type-I error, i.e., false alarm rate, controllability; and propose a computationally highly efficient online algorithm, which closely achieves a specified false alarm rate while maximizing the detection power. Regardless of whether the source is stationary or nonstationary, the proposed algorithm sequentially receives a time series and learns the nominal attributes—in the online setting—under possibly varying Markov statistics. Then, an anomaly is declared at a time instance, if the observations are statistically sufficiently deviant. Moreover, the proposed algorithm is remarkably versatile since it does not require parameter tuning to match the desired rates even in the case of strong nonstationarity. The presented study is the first to provide the online implementation of Neyman-Pearson (NP) characterization for the problem such that the NP optimality, i.e., maximum detection power at a specified false alarm rate, is nearly achieved in a truly online manner. In this regard, the proposed algorithm is highly novel and appropriate especially for the applications requiring sequential data processing at large scales/high rates due to its parameter-tuning free computational efficient design with the practical NP constraints under stationary or non-stationary source statistics.**

***Index Terms*—Anomaly detection, efficient, false alarm, Markov, Neyman-Pearson, NP, online, time series, type-I error.**

I. Introduction

DETECTION of anomalous patterns is of great interest in signal processing [1]–[4] and machine learning [5] since the irregular data due to an anomaly often detrimentally affects the target application and might even require special actions in certain scenarios [5]–[7]. For instance, a hacked computer/mobile device produces suspicious network traffic [8]; or an illegal U-turn in an intersection scene creates anomalous video data [3]; or an anomalous pattern in the electricity consumption data of a factory should definitely raise concerns [9]. Similarly, visual occlusions generate unpredictably irregular video data and reduce the object recognition rates [10]. In this paper, we study the anomaly detection problem for the temporal data in the online setting and propose a novel and computationally highly efficient online algorithm. The proposed algorithm sequentially receives a time series, learns -in an online manner- the nominal attributes in the data and detects the anomalous subsequences. We use the Neyman-Pearson (NP) characterization [11] for the anomalies and nearly achieve a constant controllable false alarm rate with maximum possible detection power regardless of whether the source is stationary or non-stationary. The proposed algorithm is able to process data in real time at extremely large scales/high rates with linear complexity in the size of the stream. Moreover, we do not require parameter tuning to match the desired rates even if the source statistics change.

There exists an extensive literature on anomaly detection [1], [3], [5]–[7] and the problem is studied under different nomenclature depending on the anomaly types such as novelty detection [2], [12], [13], outlier detection [4], [14], [15], one class learning [16], [17], intrusion detection or fault detection [18]. However, as the first time in the literature, we focus on online anomaly detection in stationary or non-stationary fast streaming temporal data with online controllability of the Type-I error that maximizes the detection power without requiring parameter tuning. Thus, the presented study considerably differs from the literature. We consider that the controllability of the false alarm rate is a crucial capability especially in the context of anomaly detection since anomalies in general draw attention and provide actionable information. In this regard, a number of false alarms more than a bearable rate is clearly frustrating and hence potentially risks the practicality of the algorithm [19], [20]. For this reason, we study the problem in the binary hypothesis framework by using the NP formulation [11], where we explicitly bound the false alarm rate (i.e. minimizing Type-I error) while achieving the maximum detection power (i.e. minimizing Type-II error). Although the NP approach is successfully applied to anomaly detection [4], [19], [20], the online implementation of the NP solution has been left unexplored. Furthermore, the existing batch NP solutions are typically based on the assumption of independent and identically distributed (i.i.d.) observations, which hardly holds in the case of temporal observations, where the data is typically highly correlated in time and non-stationary. In contrast, we model such intrinsic correlations via Markov models without assuming stationary source statistics.

Our method falls in the category of statistical anomaly detection with modeled nominal densities [5], [21], where an anomaly

Manuscript received March 30, 2015; revised August 28, 2015 and November 10, 2015; accepted November 14, 2015. Date of publication November 30, 2015; date of current version February 09, 2016. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Ruixin Niu. This work was supported in part by The Scientific and Technological Research Council of Turkey (TUBITAK) under Contract 113E517 and in part by the Turkish Academy of Sciences Outstanding Researcher Program.

H. Ozkan is with the Department of Electrical and Electronics Engineering, Bilkent University, Ankara 06800, Turkey. He is also with the UGES Division at Aselsan Inc., Ankara, Turkey (e-mail: huseyin@ee.bilkent.edu.tr; huozkan@aselsan.com.tr).

F. Ozkan is with the Department of Information Systems, Middle East Technical University, Ankara 06800, Turkey (e-mail: fatih.ozkan@metu.edu.tr).

S. S. Kozat are with the Department of Electrical and Electronics Engineering, Bilkent University, Ankara 06800, Turkey (e-mail: kozat@ee.bilkent.edu.tr).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSP.2015.2504345

is an observation that is (suspected to be) not generated by the assumed nominal stochastic model. Accordingly, we learn a parametric model (a birth-death type discrete time Markov chain in this study) for the time series data; and our anomaly detection approach is to optimally test (in the sense of NP constraints) whether the sufficient statistics of a new sequence is consistent with the learned model. A popular approach in statistical anomaly detection is to consider the distance from a suspicious instance to the nominal set of data. In [22], if the k'th minimum distance is sufficiently large, then an anomaly is declared. The rankings of these k'th distances are used in [23] to bound the number of false alarms in a computationally relatively more efficient manner. Such rankings have shown in [20] to be an asymptotically consistent estimator for inclusion in the Minimum Volume (MV) set, which is the complement of the optimal decision region for anomalies in the NP formulation when the anomalies are assumed to be uniformly distributed, cf. [19]. The method in [20] impressively avoids the explicit calculation of the MV set but rather calculates the sufficient membership indicator via the k'th rankings. Another method is the Geometric Entropy Minimization (GEM) [24], which compares a test instance with only the most concentrated subset of the nominal training data that is asymptotically convergent to the MV set.

We also use the Minimum Volume (MV) set approach to detect anomalies in temporal data. However, our goal is to obtain a computationally scalable algorithm while maintaining the NP optimality, i.e., Type-I error controllability with maximum detection power. These methods [19], [20], [22]–[24] are batch processing methods, i.e., not online, without any update strategies; and they are computationally too demanding for real time processing in fast streaming data applications. For instance, the methods [20], [22], [23] require pair-wise distance calculations and sorting to obtain the rankings and essentially to order the likelihoods, which typically results in quadratic complexity in the data size (and which are even further complex, if one also considers the model updating for the new data arriving at each time). The method in [19] is appropriate only for the batch processing and only when the batch data is of relatively small number of instances and of low dimensionality, e.g., 2-dimensional, due to the -for instance- computationally heavy and not scalable dyadic-tree implementation. Similarly, the GEM method [24] is computationally not tractable, i.e., impractical; and its tractable version (presented as another algorithm in [24]) still requires computationally heavy batch processing without the original statistical guarantees. Thus, one can hardly use such methods in our framework of sequential data processing at large scales/high rates, although they are impressive batch processing techniques. On the contrary, instead of such distance calculations and orderings, we directly approximate the distribution of the likelihoods under the general Markov models for temporal data and analytically calculate the desired quantile for the MV set. This allows a computationally highly efficient and online implementation of the NP approach with controllable Type-I error and maximum detection rate. Moreover, we do not require (like [20] and unlike others) parameter tuning to match the desired false alarm rate and also, we do not assume (unlike these methods) stationary data source.

The goal in this paper is to detect anomalous subsequences in a time series. This instance of the problem for temporal data (not in the i.i.d. batch setting like [5], [19], [20]) has also been considered, cf. [25]–[36] and the references therein. However, most studies do not address the problem in the online setting and they assume stationary source statistics [6], [7]. An anomaly score is assigned to each fixed length subsequence using the pair-wise distances among all possible such subsequences in [25] and the detection is based on the magnitude of the anomaly score. Several approaches [27]–[30] are then proposed to relieve the computational burden of the pair-wise distance calculations such as tree representations and prunings [26], local hashing [27] and Haar transform [29], [34]. Instead of the standard Euclidean distance, a compression based similarity measure is also investigated in [31], [32]. Several other methods exists, which consider unevenly sampled stochastic processes [33] as well as differently defined anomalies [7], [35], [36]. Despite the impressively efficient implementations (for instance [26], [30], [32], [36]), the model free setting along this line of research requires to investigate pair-wise relations or extract/apply complex features/transformations. Hence, their solutions are essentially not appropriate to process large scale data in real time due to their computationally heavy requirements. In contrast, we exploit the availability of the data in huge amounts and therefore the ability to precisely estimate a general Markov model, which conveniently avoids such complex computations. Additionally, unlike these discussed studies, we ensure the false alarm controllability without parameter tuning even in the case of the non-stationary sources.

Markov models are also frequently used for anomaly detection in temporal data [3], [5]–[7], [37]–[41]. The unknown Markov model parameters are first estimated in the training phase and thereafter, if the probability of a test instance is sufficiently small compared to a threshold, then an anomaly is declared in these studies. In [38], this approach is successfully demonstrated for first order Markov models, where the extension to the desired generality with higher order is straightforward. Efficient representations for large alphabet sizes is considered in via a suffix tree used in conjunction with a finite state automata in [41]; and also an extended finite state automata in [39], [40]. In [37], anomalies are defined as labeling errors in case of the hidden Markov models and their effects on state recognitions are investigated. In the video anomaly identification [3], the pixel based motion patterns are modelled under first order Markov statistics and the sufficiently unlikely patterns are labeled anomalous. Markov models are generally efficient and can be extended to online implementations. However, it is difficult to related the threshold parameters in these studies to the false alarm rates, and even once tuned correctly; re-adjustments is necessary, if the source statistics change. In contrast, our online algorithm does not require parameter tuning, i.e., it is parameter-tuning free, regardless of the stationary or non-stationary source statistics. We also guarantee to match the specified bearable false alarm rate (constant false alarm rate) while operating on the fast streaming input data in a truly online manner at the maximum possible anomaly detection power.

We provide the problem formulation in Section II. After the proposed method is described in Section III, we present our online algorithm in Section IV. We demonstrate the performance of our algorithm in Section V. The paper concludes with final remarks in Section VI.

II. Problem Formulation

We first concentrate on the stationary sources, then proceed to non-stationary settings in Section IV and present examples in Section V.

We consider a real valued, bounded, stationary and ergodic discrete time stochastic process[1] $\boldsymbol{X_t}$, i.e., $X_t \in R \subset \mathbb{R}$ with the range space R being (arbitrarily large and) finite, and $|X_t| \leq A \in \mathbb{R}^+$. Our goal is to detect anomalous subsequences, i.e., windows, in a given realization $\boldsymbol{x_t}$ from the process $\boldsymbol{X_t}$. For this purpose, we define a (sliding) window sequence $\boldsymbol{w_n} \triangleq \{w_n\}_{n=1}^{L}$ of window length L at a time t with $w_n = x_{t-L+n}$. Note that the definition of the window sequence depends on the time t, which is not shown in "$\boldsymbol{w_n}$" for notational simplicity. Then, we decide whether the sequence $\boldsymbol{w_n}$ is statistically consistent with the underlying process $\boldsymbol{X_t}$, i.e., anomalous, or not.

An anomaly often occurs due to an external factor overwriting the actual data or an abrupt change in the source statistics [5], [10]. Since the characteristics of such an external factor or a sudden change cannot be predicted beforehand, it is reasonable to assume that an anomaly can be at any point in the observation domain, i.e., it is uniformly distributed [10], [19], [20]. To detect anomalies of this kind, we formulate the problem in the Neyman-Pearson (NP) testing framework, where one minimizes the miss rate at the cost of a pre-specified false alarm rate $\tau \in [0,1]$. The NP test in this regard declares an anomaly when

$$f(\boldsymbol{w_n}) \leq \delta(\tau, L), \tag{1}$$

such that

$$\delta(\tau, L) = \max\left\{\nu : \sum_{\forall \boldsymbol{s_n}:\mathcal{L}(\boldsymbol{s_n})=L} 1_{\{f(\boldsymbol{s_n})\leq\nu\}} f(\boldsymbol{s_n}) \leq \tau\right\}, \tag{2}$$

where $f(\boldsymbol{w_n})$ is the probability mass function (p.m.f.) for a nominal window $\boldsymbol{W_n}$ from $\boldsymbol{X_t}$ and $\mathcal{L}(\boldsymbol{s_n})$ is the length of the sequence $\boldsymbol{s_n}$. We point out that:[2] if -for instance- the real-valued mother sequence $\boldsymbol{x_t}$ is allowed to span the real line, i.e., $R = \mathbb{R}$, then the summation in (2) must be replaced with an integration, which would not affect our derivations or our development. However, since we will use quantization in the amplitudes in our signal model later, it is convenient here to use a finite range space $R \subset \mathbb{R}$ and hence a summation in (2) without loss of generality. Since $\boldsymbol{w_n}$ is random, the log-likelihood $z \triangleq \log f(\boldsymbol{w_n})$ is also random with the corresponding density[3] $f_Z(z)$. Then, the same test in (1) can also be written in the log-likelihood domain as

$$z \leq \delta(\tau, L), \tag{3}$$

such that

$$\delta(\tau, L) = \max\left\{\nu : \sum_{\forall z} 1_{\{z\leq\nu\}} f_Z(z) \leq \tau\right\}.$$

This test (the Neyman-Pearson (NP) test) is the "most powerful" in the sense that it yields the highest detection rate at the false alarm level τ when the anomalies are uniformly distributed [10], [19], [20]. Our aim is to devise computationally efficient online (with linear complexity in the number of instance) NP tests for anomaly detection for time series.

III. Anomaly Detection Under Markov Statistics

The problem described in Section II requires one to determine not only the log-likelihood $z = \log f(\boldsymbol{w_n})$ but also the density $f_Z(\cdot)$ of the log-likelihood in addition to the threshold δ. For this purpose, we propose to model the underlying stochastic process $\boldsymbol{X_t}$ by a Discrete Time Markov Chain (DTMC) in Section III-A and obtain the density $f_Z(\cdot)$ in Section III-B. Then, we propose our anomaly detection methods in Section III-C, for which an efficient algorithm is presented in Section IV that sequentially operates in a truly online manner.

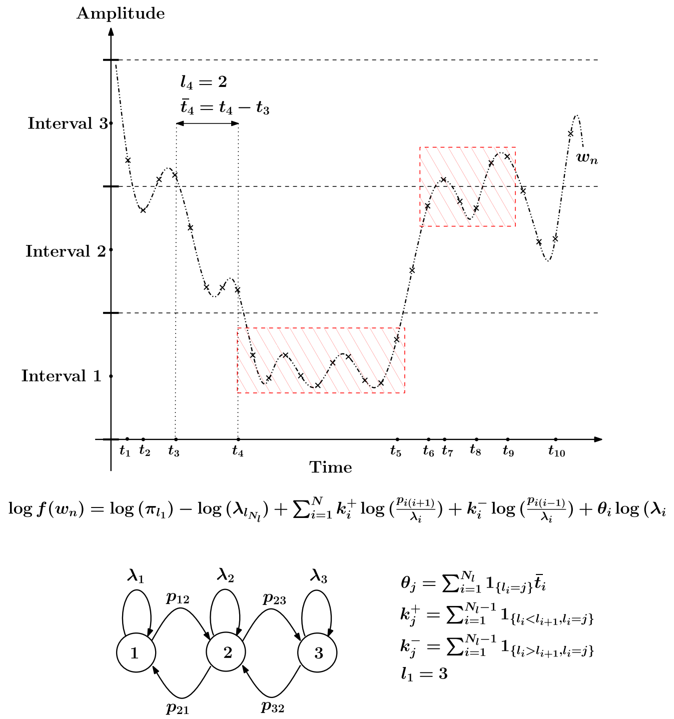
A. Observation Model

We model the unknown density $f(\cdot)$ by a Discrete Time Markov Chain (DTMC) that is assumed to govern the actual process $\boldsymbol{X_t}$, where t is the discrete time. Suppose that the range $[-A, A]$ bounding X_t is split into N intervals defining the states $1 \leq i \leq N$ of the underlying DTMC with the amplitude intervals $I_i = [-A + (i-1)\Delta, -A + i\Delta)$, where $\Delta = 2A/N$ is the length of each interval. We assume that at each time step, the process $\boldsymbol{X_t}$ preserves its state i, i.e., it stays in the corresponding amplitude interval, with probability λ_i and makes an up/down or right/left transition with probability $p_{i,i+1}/p_{i,i-1} : \lambda_i + p_{i,i+1} + p_{i,i-1} = 1$. Since the process $\boldsymbol{X_t}$ is bounded, no transitions are allowed out of the boundaries. The resulting birth-death type process is illustrated in Fig. 1. Note that for any continuous time continuous source, if it is sampled at a sufficient rate, the number N of amplitude levels in this model can always be chosen/set to obtain a birth-death type process. The reason is follows: If the sampling period is small enough, then the condition $|x_t - x_{t-1}| \leq \Delta$ is guaranteed to be satisfied for any realization $\boldsymbol{x_t}$ and at any time t (after sampling) due to continuity, which immediately yields a birth-death type process within our formulation. Also, this sampling rate affects the scale of the anomalies and our algorithm allows one to detect anomalies at any desired scale via the choice of window length L. The best choice for the window length L, which must be decided by the user, depends on, first, the target application and, second, the data through the sampling rate. For instance,

[1] Bold font types are used to indicate multiple quantities such as a sequence, vector or matrix. Upper case letters are used to indicate a matrix or random quantity. For instance, x_t is a sample, $\boldsymbol{x_t}$ is the sequence, $\boldsymbol{X_t}$ is the stochastic process; x is a scalar, X is a random variable, $\boldsymbol{x}$ is a vector and $\boldsymbol{X}$ is a matrix. Further distinction is clear from the context with no confusion. $1_{\{h\}}$ is the indicator function such that $1_{\{h\}} = 1$, if h is TRUE; and 0, otherwise. $|.|$ is the size of a set or determinant of a matrix or an absolute value depending on the argument. All listings of the form $[.]_i$ generates a column vector and $\{.\}_i$ generates a set. $\boldsymbol{X'}$ is the transpose of a matrix $\boldsymbol{X}$. $\lfloor.\rfloor$ is the floor operation for a scalar x, i.e., $\lfloor x \rfloor = \max\{y \in \mathbb{Z} : y \leq x\}$. We use the "±" notation to refer to corresponding cases respectively, i.e., $x^{\pm} = y^{\pm} : x^{+} = y^{+}$ and $x^{-} = y^{-}$. All logarithms are natural logarithms.

[2] Note that the summation in (2) is over all possible windows $\boldsymbol{s_n}$ of length L, i.e., $\mathcal{L}(\boldsymbol{s_n}) = L$; therefore, the threshold δ depends on L. Since the process $\boldsymbol{X_t}$ is stationary, the p.m.f. $f(\boldsymbol{w_n})$ accepts a common form across all windows of $\boldsymbol{X_t}$ but its form depends on the window length. Although "$f_L(\boldsymbol{w_n})$" would be a proper notation, we drop the subscript for simplification in notation.

[3] We simply use "density" or "distribution" interchangeably while referring to the probability density or mass function of a random variable. The log-likelihood $z = \log f(\boldsymbol{w_n})$ is also time varying, however, we drop the subscript for notational simplicity.


Amplitude
Interval 3
Interval 2
Interval 1
$l_4 = 2$
$\bar{t}_4 = t_4 - t_3$
w_n
t_1 t_2 t_3 t_4 t_5 t_6 t_7 t_8 t_9 t_{10}
Time
λ_1 λ_2 λ_3
p_{12} p_{23} p_{21} p_{32}
1 2 3

$$\log f(w_n) = \log(\pi_{l_1}) - \log(\lambda_{l_{N_l}}) + \sum_{i=1}^{N} k_i^+ \log(\frac{p_{i(i+1)}}{\lambda_i}) + k_i^- \log(\frac{p_{i(i-1)}}{\lambda_i}) + \theta_i \log(\lambda_i$$

$$\theta_j = \sum_{i=1}^{N_l} 1_{\{l_i=j\}} \bar{t}_i$$
$$k_j^+ = \sum_{i=1}^{N_l-1} 1_{\{l_i<l_{i+1}, l_i=j\}}$$
$$k_j^- = \sum_{i=1}^{N_l-1} 1_{\{l_i>l_{i+1}, l_i=j\}}$$
$$l_1 = 3$$

Fig. 1. A 3-state ($N = 3$) DTMC modeling of the process $\boldsymbol{X_t}$ and a corresponding window $\boldsymbol{w_n}$ is presented. Note that the sequence $\boldsymbol{x_t}$ is from an underlying signal, which is real valued and continuous; and possibly corrupted by an arbitrary (not necessarily Gaussian) and bounded additive noise. However, the noise is not illustrated in this figure for presentational clarity. The time series $\boldsymbol{w_n}$ is anomalous due to the two short-term irregularities: (i) In the boxed region on the left, it has an abnormally too long waiting time at state 1; and (ii) on the right, it has abnormally too many state transitions.

in detecting the anomalous power consumption events of the Dutch research facility [25], the desired scale of the anomalies is 1-day, and since the sampling rate is 96-readings per day, the best choice for the window length in this example is $L = 1 \times 96 = 96$.

In this study, we concentrate on real valued signals that are continuous (in the mathematical sense with respect to the time: amplitude vs time), and possibly corrupted by an arbitrary (not necessarily Gaussian) and bounded additive noise, where we use Markov chains to model the data with quantized amplitudes at discrete times. The Markov assumption here does not actually bring limitations about the assumed model and the generality is due to the Wold's Decomposition [42], which basically states that any covariance-stationary stochastic process can be decomposed as the sum of the two terms: an auto-regressive (AR) process (or equivalently a moving average (MA) process) and a deterministic mean signal. However, a direct exploitation of Wold's Decomposition leads to an infinite Markov state space, which is practically infeasible [43]. Therefore, to reduce the number of states, we partition the amplitudes into N levels, i.e., intervals. Note that although our model, i.e., the introduced birth-death type DTMC model, is a first order Markov model, the extension to higher orders is easily possible within our formulation by re-defining the history, i.e., concatenation, of sufficiently many previous states as the "new states" of a new and first order corresponding equivalent model. Regarding this straightforward extension, if one desires to use d previous observations, i.e., states, for a d'th order Markov chain, then she/he would obtain a state space of cardinality N^d with possibly a general Markov chain with almost arbitrary transitions between states violating our birth-death type chain assumption at the cost of increased computational complexity. Nevertheless, due to the almost negligible computational costs of the proposed algorithm for first order birth-death type Markov chain, one can comfortably use our algorithm at higher orders up to a certain level.

This quantization in amplitudes into N levels through the introduced DTMC modeling can also serve as an effective dimensionality reduction or an effective feature extraction step -in addition to the noise handling capabilities- with limited or no information loss. Thus, this quantization technique can actually improve the learning rates of the model to be learned by reducing the number of parameters/dimensionality and hence by mitigating the overfitting issues. For instance, the authors of [3] quantize the pixel intensity readings in time into only $N = 2$ levels for the proposed "statistical behaviour subtraction" [3]. Similarly, in terms of the Gaussian Mixtures Models (GMM) based background subtraction in video signals, N is typically around 5 [44]. Hence, we consider that quantization in amplitude is, in general, not restrictive and this quantization level N -as a design parameter- should actually be chosen as small as possible with respect to the target application while preserving information as much as possible. There is an obvious trade-off here: the assumed DTMC model might start losing from its modeling power as N gets further smaller, and N also cannot be arbitrarily large since then the noise tolerance of the assumed DTMC model decreases, i.e., it becomes more sensitive, as N gets larger. Nevertheless, this trade-off can be easily avoided by straightforwardly extending from birth-death type of Markov chains to the general Markov chains that allow arbitrary transitions between any states. Also, it is always possible to use a suitably smaller N to obtain a birth-death type chain. For instance, the non-trivial choice of $N = 2$, which always yields a birth-death type chain, has been successfully applied to video anomaly identification [3].

Furthermore, the use of the birth-death type Markov chain in our study does not cause loss of generality because our formulations can be straightforwardly extended to cover the general setting of the Markov chains with arbitrary transitions between any states and remove the birth-death type Markov assumption. On the other hand, our focus in this study is to obtain, as the first time in the literature, the online implementation of the Neyman-Pearson (NP) characterization for the anomaly detection in time series data in a truly online manner with negligible computational costs without requiring parameter tuning. To that end, as an initial study in this direction, we consider the birth-death type Markov chains, i.e., a special case of general Markov chains, that has been applied to the data with great success in a wide variety of signal processing and machine learning applications [3], [5]–[7], [44]–[49] ranging from the counting processes, e.g., queueing theory or population dynamics [45], as well as regression/classification tasks [48], [49] to video anomaly identification [3].

In our approach, we consider the small variations of a signal at any state i as insignificant variations, i.e., as contamination/noise, and hence discard them. In this sense, we do not assume that the noise is negligible. Instead, we directly handle the noise either by discarding within state variations or by allowing noisy transitions. Thus, we concentrate on the transitions (large variations)/waiting times (persistency) between/at states. In fact, within state variations become increasingly minute as the number N of the states/regions of the amplitude increases. For example, two different sequences $\boldsymbol{w}_{\boldsymbol{n}}^{(\mathbf{1})} \rightarrow \boldsymbol{y}_{\boldsymbol{n}}$ and $\boldsymbol{w}_{\boldsymbol{n}}^{(\mathbf{2})} \rightarrow \boldsymbol{y}_{\boldsymbol{n}}$ mapping to the same state sequence $\boldsymbol{y}_{\boldsymbol{n}}$, where $y_i \in \{1, 2, \ldots, N\}$ is the state of the i'th observation in $\boldsymbol{w}_{\boldsymbol{n}}^{(\boldsymbol{j})}$, are considered "same" up to small variations. In this sense, we can continue our derivations with only the state sequence $\boldsymbol{y}_{\boldsymbol{n}}$ without referring to the actual sequence $\boldsymbol{w}_{\boldsymbol{n}}$. However, to emphasize the effect of the introduced DTMC as a mapping from $\boldsymbol{w}_{\boldsymbol{n}}$ to $\boldsymbol{y}_{\boldsymbol{n}}$ and the corresponding equivalence in between, we opt to use "$\boldsymbol{w}_{\boldsymbol{n}}$" to refer to both the state sequence $\boldsymbol{y}_{\boldsymbol{n}}$ and the actual sequence $\boldsymbol{w}_{\boldsymbol{n}}$ simultaneously unless it is necessary to explicitly state the distinction. For example, $f(\boldsymbol{w}_{\boldsymbol{n}})$ is -to be more precise- the probability of the state sequence $\boldsymbol{y}_{\boldsymbol{n}}$ under the introduced DTMC model.

Based on this DTMC model, the density $f(\boldsymbol{w}_{\boldsymbol{n}})$ of a window $\boldsymbol{w}_{\boldsymbol{n}}$ from $\boldsymbol{x}_{\boldsymbol{t}}$ of finite length L is given by, cf. [45],

$$f(\boldsymbol{w}_{\boldsymbol{n}}) = \pi_{l_1} \left(\lambda_{l_{N_l}}\right)^{\bar{t}_{N_l}-1} \prod_{i=1}^{N_l-1} (\lambda_{l_i})^{\bar{t}_i-1} p_{l_i, l_{i+1}},$$

where π_{l_1} is the prior probability for the initial state l_1 that accounts for the initial conditions, $\bar{t}_i$ is the waiting time for the window $\boldsymbol{w}_{\boldsymbol{n}}$ at the state observed right before the ith transition, i.e., $\bar{t}_i = t_i - t_{i-1}$, for $1 \leq i \leq N_l$, where t_i is the time of the last observation before the i'th transition with $t_0 = 0$. Also, $\{l_i\}_{i=1}^{N_l}$ with $l_i \in \{1, 2, \ldots, N\}$ is the corresponding sequence of states observed before each transition. Note that the last transition is hypothetically assumed to be $t_{N_l} = L$ and N_l is the total number of transitions, which can be as small as only 1 transition. If we accumulate a total waiting time θ_j at state j as $\theta_j = \sum_{i=1}^{N_l} 1_{\{l_i=j\}} \bar{t}_i$, then the log-likelihood is given by

$$\log f(\boldsymbol{w}_{\boldsymbol{n}}) = \log \pi_{l_1} - \log \lambda_{l_{N_l}} + \sum_{i=1}^{N} \left(k_i^+ \log \frac{p_{i,i+1}}{\lambda_i} + k_i^- \log \frac{p_{i,i-1}}{\lambda_i} + \theta_i \log \lambda_i \right), \quad (4)$$

where $k_j = k_j^+ + k_j^-$ is the total number of state j observations in $\{l_i\}_{i=1}^{N_l}$ with the exception that $k_{l_{N_l}} = k_{l_{N_l}}^+ + k_{l_{N_l}}^- + 1$ and k_j^+ (k_j^-) is the number of "up/right" ("down/left") transitions from state j in $\boldsymbol{w}_{\boldsymbol{n}}$, i.e., $k_j^+ = \sum_{i=1}^{N_l-1} 1_{\{l_i<l_{i+1}, l_i=j\}}$ and $k_j^- = \sum_{i=1}^{N_l-1} 1_{\{l_i>l_{i+1}, l_i=j\}}$. Note that the log-likelihood expression in (4) is exact with the convention $p_{i,i+1} = p_{j,j-1} = 0 \log 0 = 0$ when $i = N$ or $j = 1$ to handle the boundary conditions.

Remark: We observe that a significant reduction via the proposed observation model is possible with no or limited information loss. Since our DTMC model is a birth-death type Markov chain, the number of up and down transitions between two states must be (almost) equal due to the Global Balance (GB) equations, i.e., $k_i^+ = k_{i+1}^- \pm 1$. Therefore, the accumulated total waiting times θ_i's at each unique state as well as the corresponding number k_i^+ of transition occurrences provide sufficient statistics and are the only signal attributes to be necessarily stored. In this sense, $\boldsymbol{d}_{\boldsymbol{w}} = \{\theta_i, k_i^+\}_{i=1}^{N}$ is a complete set of descriptors of dimensionality $2N - 1$ that is independent of the length of the sequence $\boldsymbol{w}_{\boldsymbol{n}}$.

We derive the log-likelihood density in the following Section III-B in order to later devise our anomaly detection methods in Section III-C.

B. The Log-Likelihood Density f_Z

In this part, we approximate the probability density of the log-likelihood $\log f(\boldsymbol{w}_{\boldsymbol{n}})$ in order to efficiently find the threshold of the anomaly detection test formulated in Section II. We start with concentrating on the random variables $k_i^{\pm}$, i.e., k_i^+ and k_i^-, in $\log f(\boldsymbol{w}_{\boldsymbol{n}})$; and then continue with θ_i for this derivation.

Let us define $\hat{p}_{i,i+1} = \frac{k_i^+}{\theta_i} = p_{i,i+1} + \epsilon_i^+$ and $\hat{p}_{i,i-1} = \frac{k_i^-}{\theta_i} = p_{i,i-1} + \epsilon_i^-$, in which ϵ_i^+ and ϵ_i^- are the corresponding estimation errors. Then, we can write (4) as

$$\begin{aligned} \log f(\boldsymbol{w}_{\boldsymbol{n}}) &= \log \pi_{l_1} - \log \lambda_{l_{N_l}} + \sum_{i=1}^{N} \theta_i \left(p_{i,i+1} + \epsilon_i^+\right) \log \frac{p_{i,i+1}}{\lambda_i} \\ &\quad + \theta_i \left(p_{i,i-1} + \epsilon_i^-\right) \log \frac{p_{i,i-1}}{\lambda_i} + \theta_i \log \lambda_i \\ &= \log \pi_{l_1} - \log \lambda_{l_{N_l}} + \sum_{i=1}^{N} \theta_i h_i + \sum_{i=1}^{N} \theta_i h_i^{\epsilon}, \end{aligned} \quad (5)$$

where h_i is a constant with

$$h_i = p_{i,i+1} \log \frac{p_{i,i+1}}{\lambda_i} + p_{i,i-1} \log \frac{p_{i,i-1}}{\lambda_i} + \log \lambda_i$$

and h_i^{ϵ} is an approximation term with

$$h_i^{\epsilon} = \epsilon_i^+ \log \frac{p_{i,i+1}}{\lambda_i} + \epsilon_i^- \log \frac{p_{i,i-1}}{\lambda_i}.$$

By ergodicity of the process $\boldsymbol{X}_{\boldsymbol{t}}$ and by weak law of large numbers (WLLN), $\epsilon_i^+, \epsilon_i^- \rightarrow 0$, as the length L of the sequence $\boldsymbol{w}_{\boldsymbol{n}}$ goes to infinity, i.e., $h_i^{\epsilon} \rightarrow 0$ as $L \rightarrow \infty, \forall i$ in probability. Furthermore, we point out that as $\boldsymbol{w}_{\boldsymbol{n}}$ is random, the log-likelihood $z = \log f(\boldsymbol{w}_{\boldsymbol{n}})$ is also random with the density $f_Z(z)$, to which the initial condition $\log \pi_{l_1}$ has a negligible contribution for large L. Conditioned on the knowledge of θ_i, since $\hat{p}_{i,i\pm1}$ ($\hat{p}_{i,i+1}$ and $\hat{p}_{i,i-1}$) are Maximum Likelihood (ML) estimators of the two parameters of a multinomial random variable (with three outcomes), they are unbiased with covariance $\frac{1}{\theta_i}\boldsymbol{\Sigma}_i$ and normally distributed for large θ_i, i.e., $[\epsilon_i^+, \epsilon_i^-] = \boldsymbol{\epsilon}_i | \theta_i \sim N \left(\mathbf{0}, \frac{1}{\theta_i}\boldsymbol{\Sigma}_i\right)$, where

$$\boldsymbol{\Sigma}_i = \begin{pmatrix} p_{i,i+1}(1 - p_{i,i+1}) & -p_{i,i+1} p_{i,i-1} \\ -p_{i,i+1} p_{i,i-1} & p_{i,i-1}(1 - p_{i,i-1}) \end{pmatrix}.$$

We point out that $\boldsymbol{\epsilon}_i$'s are correlated, which can be observed from the Global Balance (GB) equations from our birth-death type Markov chain, i.e., $k_i^+ = k_{i+1}^-$. Nevertheless, conditioned on the complete knowledge of $\boldsymbol{\Theta} = [\Theta_i]_{i=1}^{N}$; we naively suppose that $Z|\boldsymbol{\Theta}$ is also normally distributed for large L with mean $\mu_\theta = \mathbf{h}_\mu \boldsymbol{\theta} = \sum_{i=1}^{N} \theta_i h_i$, where $\mathbf{h}'_\mu = [h_i]_{i=1}^{N}$ (contribution of the initial and termination conditions $\log \pi_{l_1} - \log \lambda_{l_{N_l}}$ is negligible for sufficiently long observations), $\mathbf{h}'_\mu$ is the transpose of $\mathbf{h}_\mu$; and variance $\sigma_\theta^2 = \mathbf{h}_\sigma \boldsymbol{\theta}$, where

$$\mathbf{h}'_\sigma = \left[\left[\log \frac{p_{i,i+1}}{\lambda_i}, \log \frac{p_{i,i-1}}{\lambda_i} \right] \boldsymbol{\Sigma}_i \left[\log \frac{p_{i,i+1}}{\lambda_i}, \log \frac{p_{i,i-1}}{\lambda_i} \right]' \right]_{i=1}^{N}$$

(contribution of $\log \pi_{l_1} - \log \lambda_{l_{N_l}}$ is negligible), i.e.,

$$Z|\boldsymbol{\Theta} \sim f_{Z|\boldsymbol{\Theta}} = N\left(\mu_\theta, \sigma_\theta^2\right). \quad (6)$$

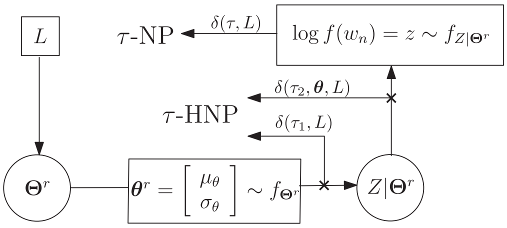

L
τ-NP
$\delta(\tau, L)$
$\log f(w_n) = z \sim f_{Z|\Theta^r}$
$\delta(\tau_2, \boldsymbol{\theta}, L)$
τ-HNP
$\delta(\tau_1, L)$
Θ^r
$\boldsymbol{\theta}^r = \begin{bmatrix} \mu_\theta \\ \sigma_\theta \end{bmatrix} \sim f_{\Theta^r}$
$Z|\Theta^r$

Fig. 2. The log-likelihood density model f_Z based on the mixture of Gaussians in (8). First the parameter $\boldsymbol{\theta^r}$ is sampled from $f_{\boldsymbol{\Theta^r}}$; then, the log-likelihood z is sampled from $f_{Z|\boldsymbol{\Theta^r}}$.

Similarly, the steady state probability π_i can be approximated [50] by the estimator $\hat{\pi}_i = \frac{\theta_i}{L} = \pi_i + \gamma_i$, which is unbiased with covariance $\frac{1}{L}\boldsymbol{\Sigma}_\gamma$, where

$$\boldsymbol{\Sigma}_\gamma = \frac{1}{L}\left\{\sum_{i=1}^{L-1}(N-i)(\mathbf{D}_{\boldsymbol{\pi}}\mathbf{P}^i + \mathbf{P}'^i\mathbf{D}_{\boldsymbol{\pi}}) + L\mathbf{D}_{\boldsymbol{\pi}}\right\} - L\boldsymbol{\pi}\boldsymbol{\pi}',$$

$\mathbf{P}$ is the matrix of transition probabilities consisting of the terms $p_{i,i\pm1}$, and $\boldsymbol{\pi} = [\pi_i]_{i=1}^N$ with $\mathbf{D}_{\boldsymbol{\pi}}$ being the diagonal matrix of $\boldsymbol{\pi}$. For sufficiently large L, $\boldsymbol{\gamma} = [\gamma_i]_{i=1}^N$ is normally distributed, i.e., $\boldsymbol{\gamma} \sim N\left(\mathbf{0}, \frac{1}{L}\boldsymbol{\Sigma}_\gamma\right)$ and hence,

$$\boldsymbol{\Theta} \sim f_{\boldsymbol{\Theta}} = N(L\boldsymbol{\pi}, L\boldsymbol{\Sigma}_\gamma). \tag{7}$$

Note that the density $f_{\boldsymbol{\Theta}}$ effectively defines a Gaussian prior on the mean μ_θ and the variance σ_θ^2 of the conditional density $f_{Z|\boldsymbol{\Theta}}$. We can further obtain a reduced set of random parameters $\boldsymbol{\theta}^r = [\mu_\theta, \sigma_\theta^2]' = \mathbf{H}\boldsymbol{\theta}$, where[4] $\mathbf{H} = [\mathbf{h}'_\mu, \mathbf{h}'_\sigma]'$ with the corresponding density $f_{\boldsymbol{\Theta}^r} = N(L\mathbf{H}\boldsymbol{\pi}, L\mathbf{H}\boldsymbol{\Sigma}_\gamma\mathbf{H}')$.

As a result of this, we obtain $Z|\boldsymbol{\Theta}^r \sim f_{Z|\boldsymbol{\Theta}^r} = N(\mu_\theta, \sigma_\theta^2)$ yielding to the (unconditional) log-likelihood density as

$$f_Z(z) = \int_{\forall\boldsymbol{\theta}^r} f_{Z|\boldsymbol{\Theta}^r}(z|\boldsymbol{\Theta}^r = \boldsymbol{\theta}^r) f_{\boldsymbol{\Theta}^r}(\boldsymbol{\theta}^r) d\boldsymbol{\theta}^r \tag{8}$$

is a mixture of Gaussians, cf. Fig. 2.

As a final remark in this section, our derivations can be straightforwardly extended to the general Markov chain with arbitrary transitions between any states (not only between the neighboring states as in the birth-death type chain) at the cost of increased computational complexity. For this purpose, the log-likelihood in (4) is first updated to incorporate the new probabilities $p_{i,j}$s of such arbitrary transitions from state i to state j with $i \neq j$ with the corresponding number of observed transitions $k_{i,j}$s. Then, the same functional log-likelihood form in (5) can be obtained with the new variables defined as $h_i = \log\lambda_i + \sum_{j=1}^N 1_{\{i\neq j\}} p_{i,j}\log\frac{p_{i,j}}{\lambda_i}$, $h_i^\epsilon = \sum_{j=1}^N 1_{\{i\neq j\}}\epsilon_{i,j}\log\frac{p_{i,j}}{\lambda_i}$, where $\epsilon_{i,j} = \frac{k_{i,j}}{\theta_i} - p_{i,j}$ is similarly the corresponding error term. Therefore, all of the log-likelihood density derivations and the Gaussian approximations in our formulation remain valid and the rest of the derivations for the extension to the general Markov chain follows similar lines. Hence, the corresponding reduced mixture of Gaussians form of (8) is straightforwardly derived in the same exact way to obtain our algorithm HNP as illustrated in Fig. 2 in the case of the general Markov chains.

C. Anomaly Detection Methods

In this section, we propose two Neyman-Pearson (NP) test based anomaly detection methods that we compare in our experiments. A) An NP test with the threshold $\delta(\tau, L)$ in (3) that is based on extensive Monte Carlo simulations named as "MCNP". B) A hierarchical NP test based on the derived mixture of Gaussians form of the log-likelihood density, named as "HNP", for which we also present a computationally highly efficient online algorithm without requiring Monte Carlo simulations or parameter tuning ("Sequential HNP") in Section IV. Regardless of whether the source is stationary or non-stationary, the method HNP successfully achieves the desired false alarm rate while maximizing the detection power.

The proposed anomaly detection method HNP is a hierarchical application of two successive NP tests: the first one uses the marginal density of $\boldsymbol{\theta^r}$ and the second one uses the log-likelihood density conditioned on $\boldsymbol{\theta^r}$, cf. Section III-C-2. During this hierarchical application, HNP allows one to analytically calculate the thresholds (of its two individual NP tests) as simple functional evaluations without requiring Monte Carlo simulations and complicated parameter tunings; and hence, yields a computationally highly efficient and online algorithm (Sequential HNP in Section IV). On the contrary, the NP test in (3) is not analytically tractable. Namely, the threshold of the NP test cannot be determined analytically, i.e., Monte Carlo simulations are necessarily used in the MCNP method. Secondly, the overall detection rate of anomalies for the test method HNP is not the best achievable when the anomalies are uniformly distributed; however, it is preferable in our study due to its efficiency. Nevertheless, the individual tests in the HNP test are all separately the most powerful as they are NP tests and the resulting HNP test is uniformly the most powerful over the variable $\boldsymbol{\theta}^r$ while achieving the desired false alarm rate τ. Moreover, when the anomalies are not uniformly distributed but appear as a change in the statistics of the underlying process $\boldsymbol{X_t}$, the optimality in the NP test is certainly lost and the proposed method HNP outperforms the method MCNP, cf. our experiments in Section V.

1) An NP Test Based on Extensive Monte Carlo Simulations (MCNP): In order to avoid the effects of the imperfect Gaussian approximation in deriving the log-likelihood density f_Z when the window length L is not sufficiently large, the first method we propose is an NP test, which is based on the exact form and the exact density of the likelihood $\log f(\boldsymbol{w_n})$ in (4); however, it heavily relies on extensive Monte Carlo simulations. We emphasize that this test is a generalization of the anomaly identification method in [3] to multi state birth-death type Markov chains and serves as a comparison basis in our experiments. Instead of approximating the density of $z = \log f(\boldsymbol{w_n})$ and calculating an approximate threshold for a specified false alarm rate τ, we estimate this threshold via extensive Monte Carlo simulations. In these simulations, we first randomly generate N_{mc} many samples of Z and obtain a set $\{z_k\}_{k=1}^{N_{mc}}$ of realizations. Note that while generating a sample z_k, we actually sample a sequence $\boldsymbol{w_n}$ of a fixed and specified window length L using the DTMC model and calculate $z_k = \log f(\boldsymbol{w_n})$ via (4). Suppose that the set $\{z_k\}_{k=1}^{N_{mc}}$ is sorted in the ascending order, i.e., $z_i \leq z_j$ for any $i \leq j$, then we estimate the true threshold in (3) as $\hat{\delta}(\tau, L) = z_{\lfloor\tau N_{mc}\rfloor}$, which precisely provides the anomaly detection method described in Section II.

[4] $\boldsymbol{X}'$ is the transpose of a matrix $\boldsymbol{X}$.

Algorithm 1: Sequential HNP

```
Input: $\boldsymbol{x_t}, L, L_e, \tau$
$\tau_1 = \tau_2 = 1 - \sqrt{1-\tau}$
while $\boldsymbol{x_t}$ is sequentially streamed; and for $t \geq L_e$ do
    $z \leftarrow z^{(t)}$ via the update rule in (14)
    $\boldsymbol{\theta} \leftarrow \boldsymbol{\theta}^{(t)}$ via the rule in (13) using the window $\boldsymbol{w_n}$
    Update the model parameters via the rule in (12)
    $\boldsymbol{\theta}^r = [\mu_\theta, \sigma_\theta^2]' \leftarrow \mathbf{H}\boldsymbol{\theta}$
    $\hat{\delta} \leftarrow C(L) + \log \tau_1$
    if $\log N(\boldsymbol{\theta^r}; L\mathbf{H}\boldsymbol{\pi}, L\mathbf{H}\boldsymbol{\Sigma}_\gamma\, \mathbf{H}') \leq \hat{\delta}$ then
        Declare anomaly at $t$
    else
        $\hat{\delta} \leftarrow \mu_\theta + \sigma_\theta Q^{-1}(\tau_2)$
        if $z \leq \hat{\delta}$ then
            Declare Anomaly at $t$
        end if
    end if
end while
Return: All found anomalies
```

2) A Hierarchical NP Test (HNP): Based on the conditional independency structure that is observed as a mixture of Gaussians in the final form of the log-likelihood density in (8), one can intuitively separate the anomaly detection problem into two pieces. Accordingly, we finally propose a hierarchical anomaly detection test method HNP that first applies an NP test for an anomaly against $\boldsymbol{\Theta}^r$ and -if not found an anomaly- secondly applies an NP test for an anomaly against $Z|\Theta^r$. We formulate this hierarchical test as

$$\begin{aligned}&\text{(a) } \log f_{\boldsymbol{\Theta}^r}(\boldsymbol{\theta}^r) \leq \delta(\tau_1, L),\\ &\text{(b) } z \leq \delta(\tau_2, \boldsymbol{\theta}^r, L),\ \boldsymbol{\theta}^r \text{ is given.}\end{aligned} \tag{9}$$

In order to ensure an overall false alarm rate τ, we also require the condition $\tau_1 + (1-\tau_1)\tau_2 \leq \tau$, where τ_1 and τ_2 are the false alarm rates of the tests for $\boldsymbol{\theta}^r$ and the conditional observation z, respectively.

Unlike the MCNP method, the HNP method requires $\boldsymbol{\theta^r}$ in addition to the log-likelihood z that is directly observable through $\mathbf{H}\boldsymbol{\theta}$ from a window sequence $\boldsymbol{w_n}$ of length L to be tested. Then, we determine the threshold $\delta(\tau_1, L)$ from

$$\tau_1 = Pr\left\{g \geq -2\hat{\delta}(\tau_1, L) + 2C(L)\right\} = e^{\hat{\delta}(\tau_1, L) - C(L)},$$

where g is exponentially distributed with mean 2 since the exponent of a bivariate Gaussian is chi-squared distributed with degree of freedom 2, and $C(L) = -\frac{1}{2}\log(4\pi^2 L^2 |\mathbf{H}\boldsymbol{\Sigma}_\gamma\, \mathbf{H}'|)$ is related to the normalization constant of a bivariate Gaussian. Hence, we obtain

$$\hat{\delta}(\tau_1, L) = C(L) + \log \tau_1. \tag{10}$$

Similarly, we obtain

$$\hat{\delta}(\tau_2, \boldsymbol{\theta}^r, L) = \mu_\theta + \sigma_\theta Q^{-1}(\tau_2), \tag{11}$$

where $Q(.)$ is the cumulative distribution function for normal distribution. Finally, the overall false alarm budget τ is to be shared between τ_1 and τ_2 such that $\tau_1 + (1-\tau_1)\tau_2 = \tau$ is satisfied to maximize the detection. This is a design issue and in this work, we set $\tau_1 = \tau_2 = 1 - \sqrt{1-\tau}$.

We point out that the method HNP does not require Monte Carlo simulations even for varying window lengths and varying source statistics since the thresholds are derived analytically. This allows a sequential and computationally highly efficient anomaly detection algorithm, cf. Section IV.

3) Estimation of the Model Parameters: Both the proposed methods MCNP and HNP require the knowledge of the model parameters $\{(\pi_i, p_{i,i\pm1}, \lambda_i)\}_{i=1}^N$ and recall that in this study, we are presented a sequence $\boldsymbol{x_t}$ and we would like to detect anomalous (sliding) windows in $\boldsymbol{x_t}$. To estimate these model parameters, we use another sliding estimation window $\boldsymbol{e_n}$ with length $L_e \gg L$, where note that $w_L = e_{L_e} = x_t$ and $\boldsymbol{e_n}$ includes $\boldsymbol{w_n}$. Then, our model parameter estimators using the signal attributes $\{\theta_i^{e_n}, k_i^{+e_n}, k_i^{-e_n}\}_{i=1}^N$ extracted from $\boldsymbol{e_n}$ are as follows:

$$\hat{p}_{i,i\pm1} = \frac{k_i^{\pm e_n}}{\theta_i^{e_n}},\ \hat{\lambda}_i = 1 - \hat{p}_{i,i+1} - \hat{p}_{i,i-1},\ \hat{\pi}_i = \frac{\theta_i^{e_n}}{L_e}, \tag{12}$$

where we use the "±" notation to refer to both cases respectively, i.e., we have two equations in (12) respectively for $\hat{p}_{i,i+1}$ and $\hat{p}_{i,i-1}$. Although the steady state distribution π_i can be analytically calculated by using the global balance equations for a birth-death type process, we also estimate it for the sake of completeness.

IV. Sequential HNP

We observe that the estimation of the model parameters, the evaluation of the log-likelihood expression in (4) and the calculation of the HNP thresholds can all be sequentially performed through a simple update. Based on this observation, we present our sequential and computationally highly efficient algorithm for the proposed anomaly detection method HNP (Sequential HNP). In the following, we describe the details of the updates for parameter learning and log-likelihood calculations that are necessary in our sequential implementation.

Parameter Learning Updates: We need to develop sequential updates for the observation dependent variables in our parameter estimation equations in (12). Suppose that we have $\{\theta_i^{(t,e_n)}, k_i^{+(t,e_n)} k_i^{-(t,e_n)}\}_{i=1}^N$ calculated based on the sequence $\boldsymbol{e_n}$ at time t; and let $y_i \in \{1, 2, \ldots, N\}$ be the state of the i'th observation in the sequence $\boldsymbol{e_n}$. After reading the instance x_{t+1}, we can update these variables as

$$\theta_i^{(t+1,e_n)} = \theta_i^{(t,e_n)} - 1_{\{y_1=i\}} + 1_{\{y_{L_e+1}=i\}}, \tag{13}$$

and $k_i^{+(t+1,e_n)}$ as well as $k_i^{-(t+1,e_n)}$ are similarly updated and we slide the parameter estimation window $\boldsymbol{e_n}$ by one step.

Log-Likelihood Updates: Suppose that we have $z^{(t)} = \log f(\boldsymbol{w_n})$, where $w_L = x_t$, and we would like to calculate $z^{(t+1)}$ with an update over $z^{(t)}$ without a re-calculation after reading the instance x_{t+1}. Let $y_i \in \{1, 2, \ldots, N\}$ be the state of the i'th observation in the sequence $\boldsymbol{w_n}$, then it is straightforward to show that

$$z^{(t+1)} = z^{(t)} + \log \frac{\hat{\pi}_{y_2}}{\hat{\pi}_{y_1}} \frac{\hat{\lambda}_{y_{L+1}}}{\hat{\lambda}_{y_1}},\ \text{if } y_1 = y_2 \text{ and } y_L = y_{L+1}, \tag{14}$$

TABLE I
AVERAGE FALSE ALARM RATES ACHIEVED BY THE HNP METHOD APPLIED TO 100 RANDOMLY INITIALIZED MARKOV MODELS OVER 5000 SEQUENCES PER EACH MODEL FOR VARYING NUMBER OF STATES N, SEQUENCE LENGTH L AND DESIRED RATES

	Desired false alarm rate τ											
	First row is for $N = 2$, the second row is for $N = 3$ and the third one is for $N = 5$											
	0.01	0.05	0.1	0.2	0.3	0.4	0.5	0.6	0.7	0.8	0.9	0.99
$L = 50$	0.011	0.048	0.093	0.180	0.268	0.364	0.462	0.570	0.682	0.816	0.932	0.995
	0.020	0.061	0.107	0.195	0.281	0.371	0.468	0.569	0.674	0.793	0.912	0.998
	0.025	0.073	0.122	0.213	0.304	0.396	0.491	0.591	0.696	0.806	0.920	0.997
$L = 100$	0.011	0.047	0.091	0.178	0.267	0.366	0.467	0.571	0.686	0.812	0.934	0.996
	0.017	0.058	0.104	0.192	0.282	0.375	0.473	0.576	0.683	0.799	0.914	0.997
	0.021	0.065	0.114	0.205	0.297	0.391	0.490	0.592	0.697	0.806	0.916	0.996
$L = 250$	0.010	0.045	0.090	0.178	0.267	0.365	0.463	0.571	0.683	0.807	0.932	0.995
	0.017	0.059	0.105	0.195	0.287	0.381	0.479	0.581	0.687	0.799	0.918	0.996
	0.016	0.059	0.106	0.199	0.293	0.389	0.488	0.591	0.697	0.805	0.916	0.995

and similarly for all of the other cases: i) $y_1 \neq y_2$ and $y_L = y_{L+1}$; ii) $y_1 = y_2$ and $y_L \neq y_{L+1}$; and iii) otherwise.

Based on these sequential likelihood and parameter updates in (13) and (14), as well as the threshold rules of HNP in (10) and (11), we present the sequential HNP in Algorithm 1.

Unlike the method MCNP, we analytically calculate the appropriate thresholds for a specified desired false alarm rate τ without using Monte Carlo simulations in our anomaly detection method HNP. This is a strong attribute of HNP that allows a generalization of our problem formulation to non-stationary sources with real time processing capabilities in a computationally highly efficient framework. Here, we assume "slow changes", i.e., "continuous drifts over time", in the source statistics to define the non-stationarity whereas an abrupt, i.e., sudden, change is used to define an anomaly, which differentiates non-stationarity from anomalies. Thus, anomaly detection in non-stationary environments is still reasonable and an effective tool. Note that the "slow" change in the source statistics refers to the "slow" rate of drift in the model parameters learned for the discrete signal obtained after the sampling of the underlying non-stationary continuous signal.

Consider a data stream $\boldsymbol{x_t}$ from a non-stationary stochastic process $\boldsymbol{X_t}$ such that its statistical properties change slowly in time; and we sequentially detect the anomalous windows of length L in $\boldsymbol{x_t}$ in real time. Our approach is to sequentially learn the time-varying model parameters in a wider and sliding window $\{e_n\}_{n=1}^{L_e}$ with $e_{L_e} = x_t$ and $L_e \gg L$ using the estimation equations in (12) and the corresponding update rules in (13), which provides the real-time adaptation to non-stationarity. Meanwhile, we apply our test at every time for windows of length L, $\{w_n\}_{n=1}^{L}$ with $w_L = x_t$ and $L \ll L_e$, using the HNP decision rule in (9) and the corresponding updates in (14). The complete algorithmic description is provided in Algorithm 1. Since our method requires only basic function evaluations and a few simple operations such as additions and subtractions, our method can be applied to stationary or non-stationary data streams with performance guarantees, i.e., NP optimality, at negligible computational costs without parameter tuning and without a re-training phase and even sequentially in a truly online manner. On the other hand, the method MCNP or the well-known other methods in the literature such as the nearest neighbor graph based detections continuously require a re-training phase, which makes them non-applicable (if one desires to maintain the crucial NP optimality) when the source is non-stationary.

Our purpose in parameter estimation in non-stationary environments is to capture, i.e., learn, an average behaviour of the time varying characteristics in the signal due to the non-stationarity, where the precise inference at each time is not reasonable, if not impossible. In order for the parameter estimation under non-stationarity to be reasonable, the variation in the model parameters, i.e., change in the source statistics, should be "slow" so that a meaningful average behaviour can be extracted and a test for an anomaly against the extracted average behaviour can be performed. Otherwise, if the non-stationarity in the signal is chaotic, then no such average behavior can be extracted and the problem itself becomes technically trivial; although the detection becomes more difficult. if the non-stationarity is chaotic, then one can observe every sequence with no surprize; and hence, one can readily use a nominal model (without a need for parameter estimation or learning) with parameters such that all possible sequences are of the same probability.

V. EXPERIMENTS

In our first set of experiments, we concentrate on the achievability of the specified false alarm rate. Note that the MCNP test method can achieve this rate arbitrarily accurately with extensive simulations. However, the proposed test HNP is designed to match the desired rate without such simulations. To this end, we devise experiments with the number of states $N \in \{2, 3, 5\}$ and the sequence length $L \in \{50, 100, 250\}$ for varying desired false alarm rates $\tau \in \{0.01, 0.05, 0.1, 0.2, \ldots, 0.9, 0.99\}$. For a given L, N and τ as well as a set of randomly chosen model parameters, i.e., $p_{i,i+1}, p_{i,i-1}, \lambda_i$, we generate 5000 length-L (normal) sequences directly from the Markov model and count the number of anomaly decisions (by HNP) that yields a false alarm rate. We report the achieved average false alarm rates over 100 randomly initialized model parameters in Table I.

We observe that the proposed method HNP approximates the desired false alarm rate more accurately for longer sequences since the Gaussianity assumption for the estimation error of the multinomial parameters also improves with the sequence length. Similarly, since the inter-state dependencies decrease as the number of states N increases, we obtain a better achievability for relatively larger N's. Finally, the proposed method HNP achieves the desired false alarm rate in most of the cases with $\pm 1\%$ error with the sequences of length $L = 250$ or longer.

We next compare the MCNP test method with the HNP test method in terms of the Receiver Operating Characteristics (ROC). In these comparisons, we concentrate on two types

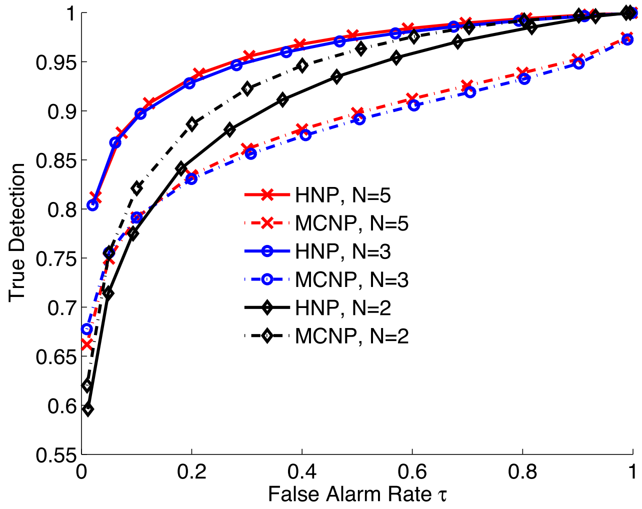


Fig. 3. The HNP test method significantly outperforms the MCNP test method when the anomalies are in terms of the abrupt model changes (cf. $N = 3$ or $N = 5$). On the other hand, when the anomalies are uniformly distributed, both methods perform comparably (cf. $N = 2$).

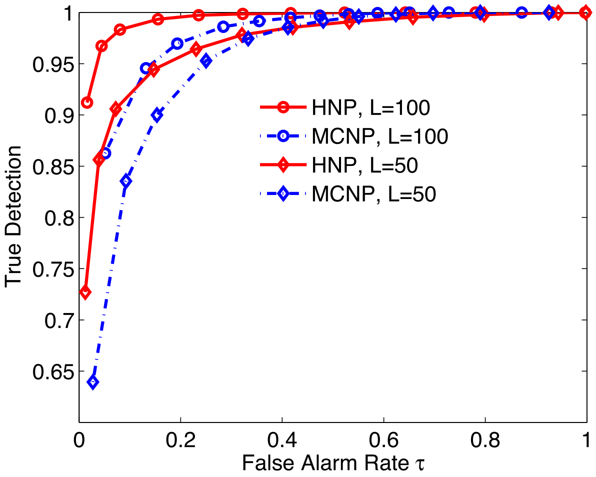


Fig. 4. The HNP test method significantly outperforms the MCNP method at almost negligible computational costs when the signal source is non-stationary.

of anomalies that are both due to a sudden change in the source statistics. In the case of the first type of anomalies, the anomalous sequences are still assumed to be drawn from a Markov model, however, whose parameters and the nominal model parameters are different. On the contrary, in the second type, anomalies do not necessarily follow a specific Markov model, where the anomalies are assumed to be uniformly and equally likely distributed in the observation domain. In this case, our purpose is to demonstrate the NP optimality since the described MCNP test is theoretically known to be optimal when the anomalies are uniformly distributed. To this end, we generate 5000 anomalous sequences of length $L = 50$, each of which is generated with respect to a randomly chosen different set of model parameters, whereas another set of 5000 normal sequences ($L = 50$) is generated with respect to a same and fixed nominal set of model parameters. In Fig. 3, we report the average ROC rates over 100 trials for varying number of states $N \in \{3, 5\}$ and desired rates τ. Secondly, we follow the same procedure, however; generate the anomalous sequences "truly uniformly" for the case $N = 2$. We observe that if the anomalies emerge as a change in the model parameters, then the proposed method HNP detects such anomalies at significantly higher rates compared to the method MCNP. Remarkably, NP optimality is observable only when the anomalies are truly uniformly distributed, cf. $N = 2$ in Fig. 3.

We emphasize that the MCNP test method heavily relies on Monte Carlo simulations to achieve the desired false alarm rate, which is prohibitively complex for real time applications. On the contrary, the proposed HNP test method is computationally highly efficient with almost negligible costs since it does not require such Monte Carlo simulations. This is a strong attribute of the HNP method, which makes it especially attractive when one needs to process non-stationary sources. We next concentrate on the truly sequential implementation of the proposed HNP method, where we perform experiments with non-stationary sources. In this part, we generate a sequence $\boldsymbol{x_t}$ of length $M = 100000$ that is specially exposed to drifting source statistics such that each instance x_t is produced based on the Markov model with the time varying parameters of $\left(\left(1 - \frac{t}{M}\right)\mathbf{P}_1 + \frac{t}{M}\mathbf{P}_2\right)$ with $N = 2$, where $\mathbf{P}_1 = \begin{bmatrix} 0 & 0.3 & 0.7 \\ 0.9 & 0 & 0.1 \end{bmatrix}$, $\mathbf{P}_2 = \begin{bmatrix} 0 & 0.05 & 0.95 \\ 0.3 & 0 & 0.7 \end{bmatrix}$. Hence, the non-stationary data in this example is generated by simulating a relatively slow change from the Markov model with the transition matrix $\mathbf{P}_1$ to the one with the transition matrix $\mathbf{P}_2$. We would like to detect the anomalous subsequences of $\boldsymbol{x_t}$ of length $L \in \{50, 100\}$ using a sliding window approach, where we use wider sliding windows of length $L_e = 10 \times L$ for estimating the active set of source statistics. We explicitly inject anomalies in $\boldsymbol{x_t}$ by overwriting the first L instances starting from every L_e'th instance of $\boldsymbol{x_t}$ by the values uniformly drawn from the support set {1,2}. We also generate a label sequence l_t, where $l_t = 1$ indicates an anomaly such that the window x_{t-L+i} includes more than $0.5 \times L$ anomalous points; and otherwise, $l_t = 0$. We run the proposed sequential HNP on randomly generated 10 different $\boldsymbol{x_t}$ sequences and report the average ROC curves in Fig. 4. On the other hand, since the signal source in this experiment is non-stationary, the threshold for the MCNP test has to be calculated at every instance via Monte Carlo simulations, which is clearly practically infeasible. Instead, for the MCNP test, we estimate the threshold only once using the complete sequence $\boldsymbol{x_t}$. We observe that the proposed HNP test method significantly outperforms the MCNP test at almost negligible computational costs with non-stationary data.

Next, we present our real data experiments, where we run our sequential HNP algorithm on the time series consisting of the power consumption readings of a Dutch research facility throughout the year 1997 [25]. These readings, i.e., power consumption measurements in 1997, are real and obtained every 15 minutes, which produces a sequence of length 96 × 365 = 35040 (96 readings per day). In this experiment, we use $N = 3$ intervals in the magnitude (the complete magnitude interval is [614,2152]) corresponding to low-[614, 1200), mid-[1200, 1600) and high-[1600, 2152] levels of power consumption. All of the choices of $N \in \{2, 3, 4, 5\}$ result in a birth-death type process as desired, however, we use $N = 3$ since it is found to be appropriate by inspection. Since this data set is provided without a ground-truth for anomalies, we inject anomalies into

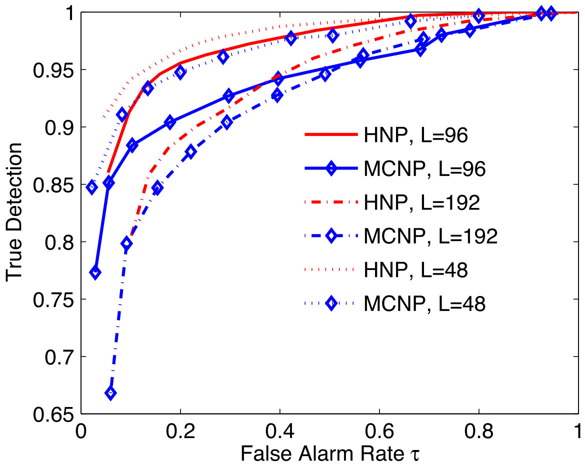


Fig. 5. Real data experiments on the power consumption data set. Our technique HNP is also robust to the mismatches between the true scale of anomalies and the guessed scale.

the data set after the quantization in order to compare (using the ROC curve) the sequential HNP (our method) and the standard NP test, i.e., MCNP, that is based on extensive Monte Carlo simulations. We inject anomalies as follows: During the first day of each month starting from February, we randomly overwrite the power consumption data (after quantization) such that a random transition from a state to another possible one (in an equally likely fashion) is applied at any time during that day. This generates $96 \times 11 = 1056$ anomalous time instances in total. We use $L_e = 96 \times 30$ (1 month) and $L \in \{48, 96, 192\}$ to also evaluate for possible mismatches between the true scale of anomalies (96 in this case) and the guessed scale (L in this case). We also use a label sequence corresponding to this sequence of power readings such that in this label sequence, any time is labeled as anomalous if the test window ending at that time has more than 50% injected anomaly instances; and labeled as nominal, otherwise. Since it is computationally prohibitive to repeatedly perform Monte Carlo simulations at every time for the method MCNP, we perform those simulations only once (for the complete stream) to calculate the desired thresholds.

Then, based on this experimental setup, we plot the ROC curve in Fig. 5 reporting the true detection rates vs false alarm rates for the methods HNP and MCNP for all cases of L (over 10 trials for injected random anomalies). We observe that our technique significantly outperforms the method MCNP in all cases due the non-stationarities in the power consumption data (for instance: months including holidays such as Good Friday or Christmas Eve consumes less power creating non-stationarity), to which the method MCNP cannot adapt (due to its computational costs), whereas our online technique HNP demonstrates excellent adaptation due to its efficient design with controllable false alarm rate. Finally, we also observe that our technique is also robust to the possible mismatches between the true scale of anomalies and the guessed scale.

VI. Conclusion

We introduce an online anomaly detection algorithm for temporal data under practical real life constraints to specifically address the contemporary applications requiring sequential data processing at large scales/high rates. The proposed algorithm is computationally highly efficient such that data streams at extremely fast rates can be processed in real time. Our algorithm also allows real time controllability of the Type-I error, i.e., false alarm rate, by nearly achieving a user specified false alarm rate while maximizing the detection power regardless of whether the source is stationary or non-stationary. Moreover, we do not require parameter tuning (to match the desired rates) even if the source statistics change. The proposed algorithm sequentially learns the possibly varying nominal Markov statistics in a time series and detects the anomalous, i.e., statistically sufficiently deviant, subsequences based on a Neyman-Pearson (NP) characterization for anomalies. The presented study is highly novel since we are the first to provide an online NP solution to the problem such that the NP optimality, i.e., maximum detection power at a specified false alarm rate, is nearly achieved in a truly online manner.

References

[1] H. Wang, M. Tang, Y. Park, and C. Priebe, “Locality statistics for anomaly detection in time series of graphs,” *IEEE Trans. Signal Process.*, vol. 62, no. 3, pp. 703–717, 2014.

[2] M. Filippone and G. Sanguinetti, “A perturbative approach to novelty detection in autoregressive models,” *IEEE Trans. Signal Process.*, vol. 59, no. 3, pp. 1027–1036, 2011.

[3] V. Saligrama, J. Konrad, and P. Jodoin, “Video anomaly identification,” *IEEE Signal Process. Mag.*, vol. 27, no. 5, pp. 18–33, 2010.

[4] J. Lehtomaki, J. Vartiainen, M. Juntti, and H. Saarnisaari, “Cfar outlier detection with forward methods,” *IEEE Trans. Signal Process.*, vol. 55, no. 9, pp. 4702–4706, 2007.

[5] V. Chandola, A. Banerjee, and V. Kumar, “Anomaly detection: A survey,” *ACM Comput. Surv. (CSUR)*, vol. 41, no. 3, p. 15, 2009.

[6] V. Chandola, A. Banerjee, and V. Kumar, “Anomaly detection for discrete sequences: A survey,” *IEEE Trans. Knowl. Data Eng.*, vol. 24, no. 5, pp. 823–839, 2012.

[7] M. Gupta, J. Gao, C. Aggarwal, and J. Han, “Outlier detection for temporal data: A survey,” *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 9, pp. 2250–2267, 2014.

[8] S. Rajasegarar, C. Leckie, and M. Palaniswami, “Anomaly detection in wireless sensor networks,” *IEEE Wireless Commun.*, vol. 15, no. 4, pp. 34–40, 2008.

[9] V. Saligrama and M. Zhao, “Local anomaly detection,” in *Proc.Int. Conf. Artif. Intell. Statist.*, 2012, pp. 969–983.

[10] H. Ozkan, O. Pelvan, and S. Kozat, “Data imputation through the identification of local anomalies,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. PP, no. 99, p. 1-1, 2015.

[11] V. Poor, *An Introduction to Signal Detection and Estimation*. New York, NJ, USA: Springer Sci. Bus. Media, 1994.

[12] V. Jumutc and J. Suykens, “Multi-class supervised novelty detection,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 36, no. 12, pp. 2510–2523, 2014.

[13] X. Ding, Y. Li, A. Belatreche, and L. Maguire, “Novelty detection using level set methods,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 3, pp. 576–588, 2015.

[14] Y. Chen, X. Dang, H. Peng, H. Bart, and H. Bart, “Outlier detection with the kernelized spatial depth function,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 31, no. 2, pp. 288–305, 2009.

[15] H. Ferdowsi, S. Jagannathan, and M. Zawodniok, “An online outlier identification and removal scheme for improving fault detection performance,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 5, pp. 908–919, 2014.

[16] B. Schölkopf, J. Platt, J. Shawe-Taylor, A. Smola, and R. Williamson, “Estimating the support of a high-dimensional distribution,” *Neural Computat.*, vol. 13, no. 7, pp. 1443–1471, 2001.

[17] B. Liu, Y. Xiao, P. Yu, L. Cao, Y. Zhang, and Z. Hao, “Uncertain one-class learning and concept summarization learning on uncertain data streams,” *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 2, pp. 468–484, 2014.

[18] C. Manikopoulos and S. Papavassiliou, “Network intrusion and fault detection: A statistical anomaly approach,” *IEEE Commun. Mag.*, vol. 40, no. 10, pp. 76–82, 2002.

[19] C. Scott and R. Nowak, “Learning minimum volume sets,” *J. Mach. Learn. Res.*, vol. 7, pp. 665–704, 2006.

[20] M. Zhao and V. Saligrama, “Anomaly detection with score functions based on nearest neighbor graphs,” *Adv. Neural Inf. Process. Syst.*, pp. 2250–2258, 2009.

[21] M. Basseville and I. Nikiforov *et al., Detection of Abrupt Changes: Theory and Application*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1993, vol. 104.

[22] K. Zhang, M. Hutter, and H. Jin, "A new local distance-based outlier detection approach for scattered real-world data," *Adv. Knowl. Discov. Data Mining*, pp. 813–822, 2009.

[23] S. Ramaswamy, R. Rastogi, and K. Shim, "Efficient algorithms for mining outliers from large data sets," *ACM SIGMOD Rec.*, vol. 29, no. 2, pp. 427–438, 2000.

[24] A. Hero, "Geometric entropy minimization (gem) for anomaly detection and localization," *Adv. Neural Inf. Process. Syst.*, pp. 585–592, 2006.

[25] E. Keogh, J. Lin, S. Lee, and H. Van Herle, "Finding the most unusual time series subsequence: Algorithms and applications," *Knowl. Inf. Syst.*, vol. 11, no. 1, pp. 1–27, 2007.

[26] E. Keogh, J. Lin, and A. Fu, "Hot sax: Efficiently finding the most unusual time series subsequence," in *Proc. IEEE Int. Conf. Data Mining*, 2005.

[27] L. Wei, E. Keogh, and X. Xi, "Saxually explicit images: Finding unusual shapes," in *Proc. Int. Conf. Data Mining*, 2006, pp. 711–720.

[28] Y. Bu, O. Leung, A. Fu, E. Keogh, J. Pei, and S. Meshkin, "Wat: Finding top-k discords in time series database," in *SDM*, 2007, pp. 449–454.

[29] A. Fu, O. Leung, E. Keogh, and J. Lin, "Finding time series discords based on Haar transform," *Adv. Data Mining Appl.*, pp. 31–41, 2006.

[30] J. Lin, E. Keogh, A. Fu, and H. Van Herle, "Approximations to magic: Finding unusual medical time series," in *Proc. IEEE Symp. Comput.-Based Med. Syst.*, 2005, pp. 329–334.

[31] E. Keogh, S. Lonardi, and C. Ratanamahatana, "Towards parameter-free data mining," in *Proc. Int. Conf. Knowl. Discov. Data Mining*, 2004, pp. 206–215.

[32] D. Yankov, E. Keogh, and U. Rebbapragada, "Disk aware discord discovery: Finding unusual time series in terabyte sized datasets," *Knowl. Inf. Syst.*, vol. 17, no. 2, pp. 241–262, 2008.

[33] X. Chen and Y. Zhan, "Multi-scale anomaly detection algorithm based on infrequent pattern of time series," *J. Computat. Appl. Math.*, vol. 214, no. 1, pp. 227–237, 2008.

[34] C. Shahabi, X. Tian, and W. Zhao, "Tsa-tree: A wavelet-based approach to improve the efficiency of multi-level surprise and trend queries on time-series data," in *Proc. Int. Conf. Scientif. Statist. Database Manage.*, 2000, pp. 55–68.

[35] L. Wei, N. Kumar, V. Lolla, E. Keogh, S. Lonardi, and R. Chotirat, "Assumption-free anomaly detection in time series," *SSDBM*, vol. 5, pp. 237–242, 2005.

[36] Y. Zhu and D. Shasha, "Efficient elastic burst detection in data streams," in *Proc. Int. Conf. Knowl. Discov. Data Mining*, 2003, pp. 336–345.

[37] H. Ozkan, A. Akman, and S. Kozat, "A novel and robust parameter training approach for HMMS under noisy and partial access to states," *Signal Process.*, vol. 94, pp. 490–497, 2014.

[38] N. Ye *et al.*, "A Markov chain model of temporal behavior for anomaly detection," in *Proc. IEEE Syst., Man, Cybern. Inf. Assur. Secur. Workshop*, 2000, vol. 166, p. 169.

[39] C. Michael and A. Ghosh, "Two state-based approaches to program-based anomaly detection," in *Proc. Ann. Conf. Comp. Secur. Appl.*, 2000, pp. 21–30.

[40] V. Chandola, V. Mithal, and V. Kumar, "Comparative evaluation of anomaly detection techniques for sequence data," in *Proc. Int. Conf. Data Mining*, 2008, pp. 743–748.

[41] C. Marceau, "Characterizing the behavior of a program using multiple-length n-grams," in *Proc. 2000 Workshop on New Secur. Paradigms*, 2001, pp. 101–110.

[42] J. Hamilton, *Time Series Analysis*. Princeton, NJ, USA: Princeton Univ. Press, 1994, vol. 2.

[43] B. Geiger, T. Petrov, G. Kubin, and H. Koeppl, "Optimal Kullback-Leibler aggregation via information bottleneck," *IEEE Trans. Autom. Control*, vol. 60, no. 4, pp. 1010–1022, 2015.

[44] Z. Zivkovic, "Improved adaptive Gaussian mixture model for background subtraction," in *Proc. IEEE 17th Int. Conf. Pattern Recogn. (ICPR)*, 2004, vol. 2, pp. 28–31.

[45] S. Karlin, *A First Course in Stochastic Processes*. New York, NY, USA: Academic, 2014.

[46] B. Moser and T. Natschlager, "On stability of distance measures for event sequences induced by level-crossing sampling," *IEEE Trans. Signal Process.*, vol. 62, no. 8, pp. 1987–1999, 2014.

[47] D. Morgan, "On level-crossing excursions of Gaussian low-pass random processes," *IEEE Trans. Signal Process.*, vol. 55, no. 7, pp. 3623–3632, 2007.

[48] S. Kozat, A. Singer, and G. Zeitler, "Universal piecewise linear prediction via context trees," *IEEE Trans. Signal Process.*, vol. 55, no. 7, pp. 3730–3745, 2007.

[49] N. Vanli and S. Kozat, "A comprehensive approach to universal piecewise nonlinear regression based on trees," *IEEE Trans. Signal Process.*, vol. 62, no. 20, pp. 5471–5486, Oct. 2014.

[50] M. Xue and S. Roy, "Spectral and graph-theoretic bounds on steady-state-probability estimation performance for an ergodic Markov chain," in *Amer. Contr. Conf.*, 2011, pp. 2399–2404.

Huseyin Ozkan received the B.Sc. degrees in electrical and electronics engineering, and mathematics from Bogazici University, Istanbul, Turkey, in 2007. He received the M.Sc. degree in electrical engineering from Boston University, MA, USA, in 2010; and the Ph.D. degree in electrical and electronics engineering from Bilkent University, Ankara, Turkey, in 2015.

He is also with the UGES Division at Aselsan Inc., Ankara, Turkey, where he conducts computer vision research for large are surveillance. Before joining Aselsan, he focused on anomaly detection and recommendation problems as a researcher at Turk Telekom Inc., Ankara. He also worked as a research intern at Mitsubishi Electric Research Laboratories, Cambridge, MA, USA, where he developed efficient algorithms for vision based road sign detection. His research interests include statistical learning, pattern recognition, computer vision and statistical signal processing.

Dr. Ozkan has been awarded the Best Paper award by the IEEE Conference on Advanced Video and Signal-based Surveillance (2011); and the Best Student Paper award by the IEEE Conference on Signal Processing Applications (2012).

Fatih Ozkan received the B.Sc. degree in computer engineering from Cukurova University, Adana, Turkey, in 2012.

He is currently working toward the M.Sc. degree in the Department of Information Systems, Middle East Technical University. His research interests include computer vision and machine learning. He is also working as a full-time researcher in the ILTAREN Institute at TUBITAK, Ankara, Turkey.

Suleyman S. Kozat (SM'12) received the B.Sc. degree in electrical and electronics engineering from Bilkent University, Ankara, Turkey. He received the M.S. and Ph.D. degrees in electrical engineering from the University of Illinois at Urbana Champaign, IL, USA, in 2001 and 2004, respectively.

After graduation, he joined IBM Research, T. J. Watson Research Center, Yorktown, NY, USA, as a Research Staff Member in Pervasive Speech Technologies Group, where he focused on problems related to statistical signal processing and machine learning. He also worked as a Research Associate at Microsoft Research, Redmond, WA, USA, in Cryptography and Anti-Piracy Group. Currently, he is an Associate Professor with the Electrical and Electronics Engineering Department, Bilkent University, Turkey. His research interests include intelligent systems, adaptive filtering for smart data analytics, online learning, and machine learning algorithms for signal processing.

Dr. Kozat has been awarded the IBM Faculty Award by IBM Research in 2011, Outstanding Faculty Award by Koc University in 2011, Outstanding Young Researcher Award by the Turkish National Academy of Sciences in 2010, ODTU Prof. Dr. Mustafa N. Parlar Research Encouragement Award in 2011, and holds the Career Award by the Scientific Research Council of Turkey, 2009.