J Sign Process Syst (2018) 90:175–184
DOI 10.1007/s11265-016-1147-0

Efficient NP Tests for Anomaly Detection Over Birth-Death Type DTMCs

Huseyin Ozkan[1] · Fatih Ozkan[2] · Ibrahim Delibalta[3] · Suleyman S. Kozat[4]

Received: 28 December 2015 / Revised: 21 April 2016 / Accepted: 24 May 2016 / Published online: 4 June 2016

Abstract We propose computationally highly efficient Neyman-Pearson (NP) tests for anomaly detection over birth-death type discrete time Markov chains. Instead of relying on extensive Monte Carlo simulations (as in the case of the baseline NP), we directly approximate the log-likelihood density to match the desired false alarm rate; and therefore obtain our efficient implementations. The proposed algorithms are appropriate for processing large scale data in online applications with real time false alarm rate controllability. Since we do not require parameter tuning, our algorithms are also adaptive to non-stationarity in the data source. In our experiments, the proposed tests demonstrate superior detection power compared to the baseline NP while nearly achieving the desired rates with negligible computational resources.

Keywords Anomaly detection · Neyman pearson · NP · False alarm · Efficient · Online · Markov · DTMC

✉ Huseyin Ozkan
hozkan@mit.edu

[1] Department of Brain and Cognitive Sciences at Massachusetts Institute of Technology, Cambridge, MA, USA

[2] Department of Information Systems at Middle East Tehnical University, Ankara, Turkey

[3] Turk Telekom Communications Services, Istanbul, Turkey

[4] Department of Electrical and Electronics Engineering at Bilkent University, Ankara, Turkey

1 Introduction

The irregular data that is significantly different from the vast majority often deserves special attention especially in the security applications [2, 3, 5, 15]. For instance, an atypical or suspicious network activity can be due to a hacked computer and should raise concerns [17]. Similarly, the failure (of a machine) can produce irregular power consumption in a factory, where the failure can be detected by a careful inspection of the data and then fixed [7]. For this reason, detection of the irregularity has been widely studied in the signal processing [4, 8, 19, 21] and machine learning [2] literatures, where the problem is referred to by various names such as anomaly detection, outlier detection and intrusion detection [2]. Since the output of the anomaly detection is in general an *alarm* requiring immediate action, an intolerable number of false alarms is certainly (and relatively more) frustrating (compared to the other type of detections) [20, 25]. To be more precise, the false alarm rate controllability in the anomaly detection applications is a crucial capability in addition to achieving the highest possible detection power. Therefore, a natural formulation of the anomaly detection problem is obtained through the Neyman-Pearson (NP) characterization [20, 25] since the NP test maximizes the detection power at a specified bearable false alarm rate [16]. Moreover, a correct detection, i.e., a true alarm, should also be produced promptly in a timely manner since, otherwise, it might be late to take the required action. To this end, we consider the temporal data produced by a birth-death type discrete time Markov chain (birth-death type DTMC) and propose two computationally efficient and highly scalable NP tests, which are appropriate for processing large scale data in real time. This study generalizes the online anomaly detection method of [14] with almost the same

computational resources. Thus, our technique proposed in this paper is also directly applicable to online applications via the sequential framework of [14].

The birth-death type DTMC is frequently used in many applications [6, 11, 12, 19, 26] for various purposes such as the learning of the statistical behavior [6, 19] and the background image [26] in video observations. Birth-death type chains are also used in [11, 12] for level-crossing based sampling. Other popular examples are in the literature of counting processes and their queuing applications, cf. [6] as well as the references therein. Hence, we emphasize that the proposed computationally highly efficient anomaly detection methods can be used to process the large scale data from a wide variety of applications in various domains. On the other hand, several impressive anomaly detection techniques have been already proposed for the data generated by the Markov chains [1–3, 5, 9, 10, 13, 19, 23], which are closely related to our study. However, these techniques do not directly and explicitly consider false alarm controllability, which is -in contrast- the main focus in this paper. For example, a test instance is declared anomalous in [23] if its probability under the constructed first order Markov model is below a certain threshold, where Monte Carlo simulations are required to relate this probability threshold to a desired false alarm rate. Such a requirement makes the method [23] impractical for large scale applications, whereas our technique maximizes the detection power at the desired false alarm rate with almost negligible computational costs through simple function evaluations without extensive simulations. Hence, one can directly apply our methods to fast streaming data in the, for example, online systems by using the online framework of [14]. The hidden Markov model is extended in [13] to cover the partially observed but erroneous, i.e., anomalous, state observations, which -however- does not explicitly detect anomalies but instead implicitly handle them for state recognition purposes. In the video anomaly identification method of [19], a 2-state (hence a birth death-type) Markov chain is applied to model the foreground and background labels at each pixel, where statistically significant deviations are declared as anomalous labels. The result is a statistical background being capable of anomaly detection. Similar to [23], this method [19] also heavily relies on Monte Carlo simulations. It is computationally intractable to perform [19] at each pixel in real time under varying source statistics; on the contrary, one can readily use our technique for video anomaly identification in the framework of [19] under non stationarity.

Neyman-Pearson (NP) tests for anomaly detection purposes have been successfully applied to data in several domains in machine learning [18, 20, 24, 25]. In these methods, the core approach is to declare a test instance anomalous, if the distance from the test instance to a nominal set of data is sufficiently large [24], for which an efficient implementation can be found in [18]. The generalization from this core approach to the NP characterization is through ranking the distances from a validation set of instances to the training set [25]. When the anomalies are uniformly distributed, the NP test is equivalent to simply deciding whether a test instance is in the minimum volume (MV) set of the nominal distribution [20]. Since these methods [18, 20, 24, 25] are non-parametric, the MV membership decision requires pair-wise distance calculations and ranking, which makes them computationally prohibitive for large scale applications. In this paper, we also use the NP approach by assuming that the anomalies are uniformly distributed, i.e., we also use the MV approach, to control the number of false alarms while maximizing the detection power. However, unlike non-parametric methods in [18, 20, 24, 25], our technique relies on a parametric model, i.e., birth-death type DTMC, which allows computationally highly scalable and efficient implementations through the introduced log-likelihood density approximations. Moreover, our technique does not require to relate algorithmic parameters to the desired false alarm rate, i.e., it is parameter tuning free in terms of false alarm rate controllability, with strong adaptation capabilities to non-stationarity, cf. [14] for the details of this adaptation.

We provide the problem description in Section 2. After our log-likelihood density approximation is described in Section 3, we present our efficient anomaly detection tests in Section 4. We demonstrate the performance of our algorithms in Section 5 through several numerical examples and simulations. The paper concludes with final remarks in Section 6.

2 Problem Description

Suppose that X_n (with $X_n \in \mathcal{S}, \forall n$) is a stochastic process from a birth-death type Discrete Time Markov Chain (DTMC), where one observes transitions only between the neighboring states. In this model, $\mathcal{S} = \{1, 2, \cdots, N\}$ is the state space of cardinality N (N is the number of states); and the states i and j are neighbors, if $|i - j| \leq 1$. We denote the state transition probabilities by λ_{ij} and the initial state probabilities by π_i. To be more precise, X_n starts a realization w_n from a state $j \in \mathcal{S}$ with probability π_j; preserves -once a realization is started- its state $i \in \mathcal{S}$ with probability λ_{ii} and make a transition from the state i to the state j with probability λ_{ij} at any time n. However, since this is a birth-death type DTMC, only up/down or right/left transitions are possible, i.e., $\lambda_{ij} = 0$, if $|i - j| > 1$. Note that for a given realization w_n, one can straightforwardly calculate the probability $p(w_n)$ under this model. A corresponding 4-state ($N = 4$) birth-death type DTMC is illustrated in Fig. 1.

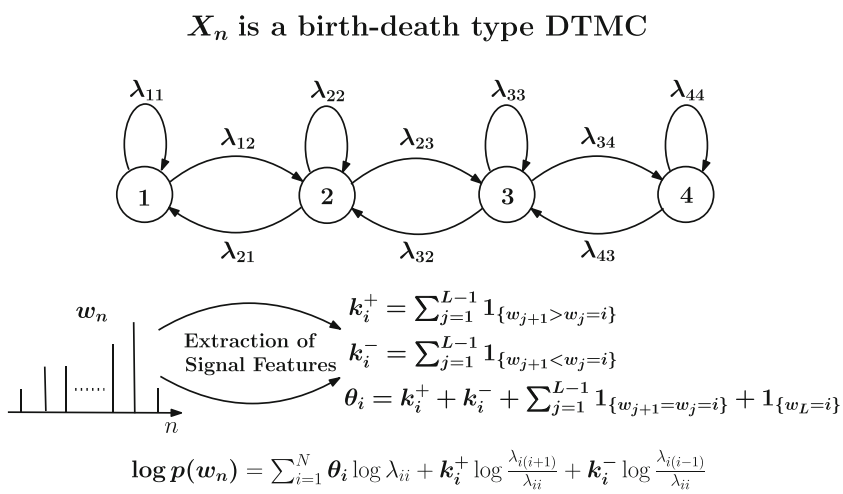

Figure 1 A 4-state ($N = 4$) birth-death type Discrete Time Markov Chain (DTMC) model is illustrated. A corresponding realization w_n (from the underlying process X_n) is processed to extract the sufficient statistics for our anomaly detection purposes such as θ_j, k_j^+, k_j^-. Here, k_i^+ (k_i^-) is the total number of right/up (*left/down*) transitions in the window sequence w_n; and θ_i is the total number of state i observations. These statistics (i.e. θ_j, k_j^+, k_j^-) are sufficient to obtain the probability $p(w_n)$ of the sequence w_n under the introduced DTMC model.

In this paper, our aim is to process big amounts of data from an infinite stream x_n of the underlying stochastic process X_n (window-by-window via a sliding window approach) and decide whether a window w_n of length L extracted from x_n is statistically consistent with the underlying process X_n, i.e., whether it is "anomalous", at a specified false alarm rate $\tau \in [0, 1]$ in a computationally highly efficient manner.

To this end, we design a test, which declares an anomaly when

$$p(w_n) \leq \delta(\tau, L) \text{ such that } \sum_{\forall w_n \text{ of length } L} p(w_n) 1_{\{p(w_n) \leq \delta(\tau, L)\}} \leq \tau, \tag{1}$$

where the second inequality is for the required false alarm rate constraint. Note that $\delta(\tau, L)$ is the detection threshold as a function of the test window length L and the user specified desired false alarm rate τ; and $1_{\{\cdot\}}$ is the indicator function returning 1 if its argument is TRUE; and returning 0, otherwise.

We emphasize that the test presented in Eq. 1 is precisely the Neyman-Pearson (NP) test, i.e., the most powerful test at the specified false alarm rate τ, when the anomalies are uniformly distributed and the threshold $\delta(\tau, L)$ is set correctly.

To obtain this most powerful NP test (under the assumption of uniformly distributed anomalies), we first re-write the same anomaly detection test in Eq. 1 through the log-likelihood transformation $z = \log p(w_n)$ such that an anomaly is declared when

$$z \leq \delta(\tau, L). \tag{2}$$

Here, we use the same notation for the detection thresholds before and after the log-likelihood transformation, which are actually different, for the presentational clarity. Also, recall that the false alarm rate due to the log-likelihood thresholding with any given threshold ν is given by

$$\sqcup(\nu) = \sum_{\forall z} p(z) 1_{\{z \leq \nu\}}.$$

Then, we find the threshold $\delta(\tau, L)$ in Eq. 2 by maximizing the anomaly detection power, i.e., minimize the miss probability, at the specified false alarm rate τ. Since the most powerful test is simply a likelihood ratio test (which becomes a simple thresholding on the log-likelihood when the anomalies are uniformly distributed), maximizing over ν maximizes the detection rate due to the NP characterization of anomalies. Therefore, the desired threshold is obtained via

$$\delta(\tau, L) = \max\{\nu : \sqcup(\nu) \leq \tau\}. \tag{3}$$

Based on this problem description, our goal is to develop the NP test, i.e., the anomaly detection rule in Eq. 2 performed with the threshold in Eq. 3 which yields the highest detection at the desired false alarm rate, in a computationally highly efficient manner under the introduced birth-death type DTMC model. Therefore, the proposed anomaly detection tests are appropriate for processing data at large scales in online applications and can be directly used in the sequential framework of [14].

3 Log-Likelihood Under DTMC

We formulate the anomaly detection problem (in Section 2) in the Neyman-Pearson (NP) framework to achieve the real time false alarm rate controllability. The presented test in Eq. 2 requires one to calculate the test statistic z, i.e., the log-likelihood $z = \log p(w_n)$ under the introduced N-state birth-death type Discrete Time Markov Chain (DTMC), as well as the corresponding test threshold $\delta(\tau, L)$. Both of these calculations must be performed computationally efficiently in order to obtain the anomaly detection at the specified false alarm rate, which is intended to be appropriate for processing data at large scales in online applications. Although the calculation of the log-likelihood $z = \log p(w_n)$ (or $p(w_n)$) is straightforward and simple, i.e., not computationally demanding, the calculation of the test threshold $\delta(\tau, L)$ is cumbersome since it requires to find the complicated log-likelihood density $p(z)$.

We note that since the window w_n of length L is from a domain of cardinality N^L, i.e., since w_n is a discrete random variable with finite sample space, the log-likelihood distribution $p(z)$ (as well as the actual distribution $p(w_n)$) is actually a probability mass function, which can be calculated as

$$p(z) = \sum_{\forall w_n : z = \log p(w_n)} p(w_n)$$

once $p(w_n)$ can be calculated. However, this indirect definition or calculation of $p(z)$ yields a complicated false alarm rate definition as

$$\sqcup(\nu) = \sum_{\forall z} \sum_{\forall w_n : z = \log p(w_n)} p(w_n) 1_{\{z \le \nu\}},$$

which in turn complicates the calculation of the correct threshold in Eq. 3 and therefore hinders the efficient implementations of the described anomaly detection.

To overcome this difficulty, we propose to approximate the log-likelihood density $p(z)$ as a mixture of Gaussian densities and directly obtain, based on this approximation, the test threshold $\delta(\tau, L)$ as the corresponding Gaussian quantiles via simple function evaluations. We explain the details of our approach in the following.

3.1 Log-Likelihood Density Approximation

Under the introduced birth-death type DTMC model, the probability of a window sequence w_n of length L (extracted from a mother sequence x_n during a window-by-window processing) can be calculated as

$$p(w_n) = \pi_{w_1} \prod_{i=1}^{L-1} \lambda_{w_i w_{i+1}},$$

where the multipliers, $\lambda_{w_i w_{i+1}}$'s, can attain only $3N - 2$ unique values with N being the number of states (since this is a birth death type chain; it would be N^2 in the general case of Markov chains), i.e., $3N - 2 = |\{\lambda_{ij} : 1 \le i \le N, 1 \le j \le N, |i - j| \le 1, \}|$.[1] Based on this observation, one can re-write the probability $p(w_n)$ as

$$p(w_n) = \pi_{w_1} \prod_{i=1}^{N} \lambda_{ii}^{k_i^o} \lambda_{i(i+1)}^{k_i^+} \lambda_{i(i-1)}^{k_i^-}, \tag{4}$$

where k_i^o is the total number of the state preservations at the state i and k_i^+ (k_i^-) is the total number of right/up (left/down) transitions in the window sequence w_n, i.e.,

$$k_i^+ = \sum_{j=1}^{L-1} 1_{\{w_{j+1} > w_j = i\}}, \; k_i^- = \sum_{j=1}^{L-1} 1_{\{w_{j+1} < w_j = i\}} \text{ and}$$

$$k_i^o = \sum_{j=1}^{L-1} 1_{\{w_{j+1} = w_j = i\}}$$

with the convention $k_N^+ = \lambda_{N(N+1)} = k_1^- = \lambda_{10} = 0$. Noting that

$$k_i^o + k_i^+ + k_i^- + 1_{\{w_L = i\}} = \theta_i$$

is the total number of state i observations in the window sequence w_n, i.e., total waiting time in the state i, one can reach

$$p(w_n) = \frac{\pi_{w_1}}{\lambda_{w_L w_L}} \prod_{i=1}^{N} \lambda_{ii}^{\theta_i} \left(\frac{\lambda_{i(i+1)}}{\lambda_{ii}}\right)^{k_i^+} \left(\frac{\lambda_{i(i-1)}}{\lambda_{ii}}\right)^{k_i^-}, \tag{5}$$

where $\frac{\pi_{w_1}}{\lambda_{w_L w_L}}$ is necessary for handling the boundary conditions. Thus, taking the logarithm of the both sides, we obtain the log-likelihood expression as

$$\log p(w_n) = \sum_{i=1}^{N} \theta_i \log \lambda_{ii} + k_i^+ \log \frac{\lambda_{i(i+1)}}{\lambda_{ii}} + k_i^- \log \frac{\lambda_{i(i-1)}}{\lambda_{ii}}, \tag{6}$$

where we omit the term (due to the initial conditions) $\log \frac{\pi_{w_1}}{\lambda_{w_L w_L}}$ since its contribution to the log-likelihood is negligible for relatively large sequence length L.

Remark 1 We emphasize that the log-likelihood $\log p(w_n)$ is a function of the three quantities, i.e., the total number θ_i of state i observations as well as the total number of right/up and left/down transitions. Thus, these are essentially the signal features that are necessary and sufficient for statistical inferences regarding the window sequence under the introduced birth-death type DTMC. Accordingly, in the proposed anomaly detection method, these are the only quantities to be extracted and processed, which actually allows our

[1] $|\cdot|$ is either the cardinality of a set or the absolute value of a scalar or the determinant of a matrix, which is understood from the argument.

computationally efficient tests that are appropriate for large scale applications, cf. Fig. 1.

After we obtain the log-likelihood expression $z = \log p(w_n)$ in Eq. 6, we next model the log-likelihood density $p(z)$ (note that since w_n is random, $z = \log p(w_n)$ is also random) in order to obtain the threshold $\delta(\tau, L)$ of the desired NP test in Eq. 2 in a computationally efficient manner such that -for example- an online implementation is possible. For this purpose, we consider the Bernoulli parameter estimation for the described sufficient statistics of the log-likelihood in Eq. 6. Note that $\frac{k_i^+}{\theta_i} = \lambda_{i(i+1)} + \epsilon_i^+$, where ϵ_i^+ is the error term, which is zero mean Gaussian distributed (conditioned on sufficiently large θ_i) with a variance approaching to 0 as θ_i increases. In the general case, the random vector $[\epsilon_i^+, \epsilon_i^-]|\theta_i = \boldsymbol{\epsilon}_i|\theta_i \sim N\left(\mathbf{0}, \frac{1}{\theta_i}\Sigma_{\mathbf{i}}\right)$, where

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

Based on these Bernoulli estimators $\frac{k_i^+}{\theta_i} = \lambda_{i(i+1)} + \epsilon_i^+$ as well as $\frac{k_i^-}{\theta_i} = \lambda_{i(i-1)} + \epsilon_i^-$, we define another scalar random variable h_i^ϵ as

$$h_i^\epsilon = \epsilon_i^+ \log \frac{\lambda_{i(i+1)}}{\lambda_{ii}} + \epsilon_i^- \log \frac{\lambda_{i(i-1)}}{\lambda_{ii}},$$

which is -conditioned on θ_i- Gaussian distributed with mean 0 and variance $\frac{1}{\theta_i}v_i$, i.e., $h_i^\epsilon|\theta_i \sim N\left(0, \frac{1}{\theta_i}v_i\right)$, where

$$v_i = \left[\log \frac{\lambda_{i(i+1)}}{\lambda_{ii}}, \log \frac{\lambda_{i(i-1)}}{\lambda_{ii}}\right] \boldsymbol{\Sigma}_i \left[\log \frac{\lambda_{i(i+1)}}{\lambda_{ii}}, \log \frac{\lambda_{i(i-1)}}{\lambda_{ii}}\right]^T.$$

Then, one can obtain

$$\log p(w_n) = z = \sum_{i=1}^{N} \theta_i h_i + \sum_{i=1}^{N} \theta_i h_i^\epsilon,$$

where h_i is a constant with

$$h_i = \lambda_{i(i+1)} \log \frac{\lambda_{i(i+1)}}{\lambda_{ii}} + \lambda_{i(i-1)} \log \frac{\lambda_{i(i-1)}}{\lambda_{ii}} + \log \lambda_{ii}.$$

Then, for the log-likelihood density conditioned on the knowledge of $\boldsymbol{\theta} = [\theta_1, \theta_2, \cdots, \theta_N]^T$ (composing of sufficiently large θ_i's), we have

$$f(z|\boldsymbol{\Theta} = \boldsymbol{\theta}) = N\left(\sum_{i=1}^{N} \theta_i h_i, \sum_{i=1}^{N} \theta_i v_i\right), \tag{7}$$

where we naively assume that ϵ_i's are independent.

Remark 2 We approximate the probability mass function $p(z|\boldsymbol{\Theta} = \boldsymbol{\theta})$ ($z|\boldsymbol{\Theta}$ is discrete) by the probability density function $f(z|\boldsymbol{\Theta} = \boldsymbol{\theta})$ ($z|\boldsymbol{\Theta}$ becomes artificially continuous for the approximation), and hence, we switch from notation "p" to "f" to make the distinction clear. Therefore, $Z|\boldsymbol{\Theta}$ is normally distributed with mean $\sum_{i=1}^{N} \theta_i h_i$ and variance $\sum_{i=1}^{N} \theta_i v_i$ for sufficiently large window length L, i.e., for any $\boldsymbol{\Theta}$ consisting of sufficiently large θ_i's.

As a result of this, we obtain the (unconditional) log-likelihood density using the priors over $\boldsymbol{\Theta}$ as

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

where the prior $f_{\boldsymbol{\Theta}}(\boldsymbol{\theta})$ is also Gaussian distributed, i.e., the probability mass function $p(\boldsymbol{\theta})$ can also be approximated by a Gaussian distribution as in the case of the conditional log-likelihood density. To obtain this prior $f_{\boldsymbol{\Theta}}(\boldsymbol{\theta})$, we again consider the Bernoulli estimator $\frac{\theta_i}{L} = \pi_i + \gamma_i$, where $\boldsymbol{\gamma} \sim N(0, \frac{1}{L}\boldsymbol{\Sigma}_\gamma)$; and therefore,

$\boldsymbol{\Theta} \sim f_{\boldsymbol{\Theta}} = N(L\boldsymbol{\pi}, L\boldsymbol{\Sigma}_\gamma)$ for sufficiently large L.

Here, the covariance $\boldsymbol{\Sigma}_\gamma$ can be straightforwardly obtained, cf. [22] for the details, as

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

where $\mathbf{R}$ is the matrix of state transition probabilities consisting of the terms $\lambda_{i(i\pm1)}$, and $\boldsymbol{\pi} = [\pi_1, \pi_2, \cdots, \pi_N]^T$ with $\mathbf{D}_{\boldsymbol{\pi}}$ being the diagonal matrix of $\boldsymbol{\pi}$ of appropriate size.

4 Anomaly Detection Tests with False Alarm Rate Controllability

We emphasize that the NP test for anomaly detection presented in Eq. 1 or equivalently (due to the log-likelihood transformation) in Eq. 2 can be readily obtained via extensive Monte Carlo simulations which is named as the "MCNP" (Monte Carlo Neyman Pearson) method in this paper, cf. Fig. 2. To be more precise, one can easily sample many window sequences $\{w_n^i\}_{i=1}^{MC}$ (of length L) and for each of these window sequences, the corresponding log-likelihood values can be calculated as $\{z_i\}_{i=1}^{MC}$ under the introduced birth-death type DTMC, where $z_i = \log p(w_n^i)$. Here, suppose that these values are sorted in the ascending order without loss of generality, i.e., $z_i \leq z_j$, if $i \leq j$. Then, the corresponding threshold in Eq. 2 can be found as $\delta(\tau, L) = z_{\lfloor \tau \times MC \rfloor}$.[2] On the other hand, due to these computationally heavy and extensive Monte Carlo simulations (the generation of sorted z_i's here) which are necessary to match the desired false alarm rate τ, MCNP is prohibitively complex for applications requiring data processing at large scales and therefore it cannot be used -for instance- in the online applications.

[2] $\lfloor \cdot \rfloor$ is the floor operation.

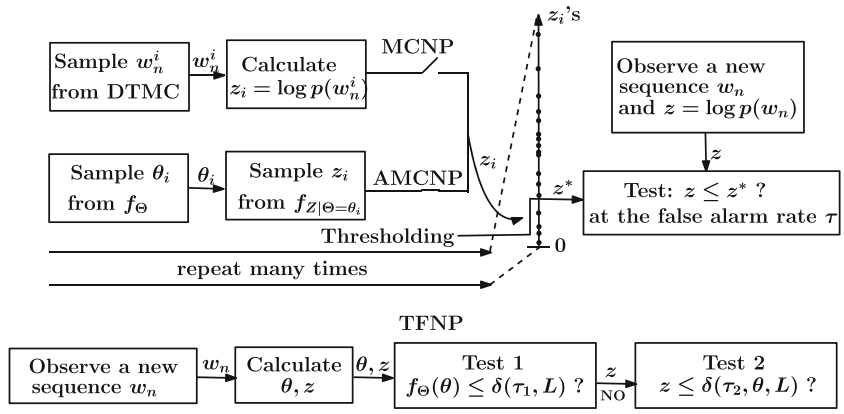

Figure 2 We propose two computationally highly efficient anomaly detection tests at the specified false alarm rate τ with real time data processing capabilities, which are based on our log-likelihood density approximation: i) AMCNP and ii) TFNP. We compare these tests with the baseline method MCNP, please refer to our experiments for the comparisons.

Nevertheless, the approximate log-likelihood density in Eq. 8 allows us to propose two other anomaly detection tests: i) The "AMCNP" (Approximate Monte Carlo Neyman Pearson) test, which is an approximate NP test through significantly more efficient Monte Carlo simulations (compared to the ones in the case of MCNP) ii) The "TFNP" (Two Fold Neyman Pearson) test which is a two-fold NP test that matches the desired false alarm rate via simple function evaluations without Monte Carlo simulations. We illustrate these tests AMCNP and TFNP in Fig. 2 and present the details in the following.

4.1 The Test Method AMCNP

This test method AMCNP is based on Monte Carlo simulations, where the execution of these simulations are computationally highly efficient compared to the simulations in the case of the test method MCNP. Namely, both the methods AMCNP and MCNP contain sampling. However, MCNP samples sequences, whereas AMCNP samples certain variables such as the total counts. The reason for the sampling efficiency in the case of AMCNP is that sampling the sequences in a generative manner (as in the case of MCNP) is computationally much more expensive than directly sampling a few variables such as the total counts (as in the case of AMCNP). Thus, AMCNP is appropriate for online applications.

AMCNP uses the approximate log-likelihood density in Eq. 8 to generate the sorted log-likelihood values $\{z_i\}_{i=1}^{MC}$ and set $\delta(\tau, L) = z_{\lfloor \tau \times MC \rfloor}$. Here, each z_i is obtained as follows: i) First, one samples θ_i from the distribution f_Θ, ii) and then samples z_i from the distribution $f_{Z|\Theta}$. Note that this described process of the generation of z_i does not require the generation of a window sequence (of length L) w_n^i (and therefore it does not require to run a birth-death type DTMC) and calculate the log-likelihood $z_i = p(w_n^i)$. Instead, since this described process only requires two simple probabilistic look-up's in addition to the "sorting" (sorting is common to both MCNP and AMCNP), it is appropriate for real-time processing of large scale data. An illustration of this test method AMCNP is presented in Fig. 2.

4.2 The Test Method TFNP

We next propose a two-fold NP test named "TFNP", which does not require extensive Monte Carlo simulations or sorting as in the cases of MCNP or AMCNP. Instead, the test method TFNP directly calculates the corresponding quantile to match the desired false alarm rate via two separate and hierarchical tests. Namely, TFNP is a successive application of two NP tests. The first test is applied against the variable $\boldsymbol{\Theta}$, i.e., the total waiting time features extracted from the window sequence w_n, at the false alarm rate τ_1. If found an anomaly, it is declared; otherwise, the second test is applied against the conditional log-likelihood $Z|\boldsymbol{\Theta} = \boldsymbol{\theta}$ at the false alarm rate τ_2 such that a desired rate τ is achieved in the end. The test method TFNP is illustrated in Fig. 2. To be more precise, we first apply the Test 1, which is designed as

$$\text{(Design of Test 1)} \quad (9)$$
$$\log f_{\boldsymbol{\Theta}}(\boldsymbol{\theta}) \leq \delta(\tau_1, L) \text{ such that}$$
$$\int_{\forall \theta} f_{\boldsymbol{\Theta}}(\boldsymbol{\theta}) 1_{\{\log f_{\boldsymbol{\Theta}}(\boldsymbol{\theta}) \leq \delta(\tau_1, L)\}} \leq \tau_1$$

and if the result is positive, i.e., $\log f_{\boldsymbol{\Theta}}(\boldsymbol{\theta}) \leq \delta(\tau_1, L)$ is TRUE, then we declare an anomaly. Otherwise, we apply the Test 2, which is designed as

$$\text{(Design of Test 2)} \quad (10)$$
$$z \leq \delta(\tau_2, \boldsymbol{\theta}, L) \text{ such that}$$
$$\int_{\forall z} f_{Z|\boldsymbol{\Theta}=\boldsymbol{\theta}}(z) 1_{\{z \leq \delta(\tau_2, \boldsymbol{\theta}, L)\}} \leq \tau_1,$$

and if the result is positive, i.e., $\log f_{\boldsymbol{\Theta}}(\boldsymbol{\theta}) \leq \delta(\tau_1, L)$ is TRUE, then we declare an anomaly. Otherwise, we declare no anomaly. We also require $\tau_1 + (1 - \tau_1)\tau_2 = \tau$ to match the overall desired false alarm rate τ.

Note that in the successive application of the individual tests of the method TFNP, the calculation of the signal

feature $\boldsymbol{\theta}$ (necessary for the Test 1) as well as the calculation of the log-likelihood z (necessary for the Test 2) for a given sequence w_n are both straightforward and can be readily performed efficiently in a truly online manner during a window-by-window processing (generation of w_n's) of a mother sequence x_n. However, the important step is the calculation of the correct thresholds $\delta(\tau_1, L)$ and $\delta(\tau_2, \boldsymbol{\theta}, L)$, to match the desired false alarm rate τ in the end. In the following, we explain the details of the calculation of these thresholds.

Let us first start with $\delta(\tau_1, L)$ and observe that

$$\log f_{\boldsymbol{\Theta}}(\boldsymbol{\theta}) = C(L) - \frac{1}{2}\alpha \leq \delta(\tau_1, L)$$

is equivalent to

$$\alpha \geq 2(C(L) - \delta(\tau_1, L)), \text{ where}$$

$C(L) = -\frac{N}{2}\log 2\pi - \frac{1}{2}\log|\boldsymbol{\Sigma}_\gamma|$, $\alpha = (\boldsymbol{\theta} - L\boldsymbol{\pi})^T \boldsymbol{\Sigma}_\gamma^{-1}(\boldsymbol{\theta} - L\boldsymbol{\pi})$ and α is chi-squared distributed with degree N. Note that the covariance $\boldsymbol{\Sigma}_\gamma$ is actually not full rank, i.e., its rank is $N-1$, since sum of θ_i's is known to be L. This condition can be removed by considering the reduced version to the rank $N-1$. We use the reduced version (via skipping the last element in $\boldsymbol{\theta}$) in this study to ensure the full rank condition. In general, one can use any reduction to the rank $N-1$ by choosing any linearly independent set of $N-1$ directions and applying the corresponding transformation, where this choice of reduction does not affect our derivations as long as the desired transformation is invertible. Hence, we need to satisfy

$$Pr(\alpha \geq 2(C(L) - \delta(\tau_1, L))) = \tau_1$$

to match the desired rate τ_1 in the first level of TFNP. Since the solution to the equation

$$1 - Pr(\alpha \leq 2(C(L) - \delta(\tau_1, L))) = \tau_1$$

yields

$$\chi(1-\tau_1, N) = 2(C(L) - \delta(\tau_1, L)),$$

we obtain

$$\delta(\tau_1, L) = C(L) - \frac{\chi(1-\tau_1, N)}{2}, \tag{11}$$

where $\chi(\cdot, N)$ is the inverse of the cumulative distribution function for the chi-squared distribution of degree N. We emphasize that the evaluations of $\chi(\cdot, N)$ do not require computational resources since it is based on a look-up table which can be prepared off-line. This completes the design of Test 1 of the TFNP method.

Our derivations of the other threshold $\delta(\tau_2, \boldsymbol{\theta}, L)$ that is used in the Test 2 of the TFNP method follows similar lines. Since

$$Pr(z \leq \delta(\tau_2, \boldsymbol{\theta}, L)) = \tau_2$$

yields

$$\tau_2 = Q\left(\frac{\delta(\tau_2, \boldsymbol{\theta}, L) - \sum_{i=1}^N \theta_i h_i}{\sum_{i=1}^N \theta_i v_i}\right),$$

we obtain

$$\delta(\tau_2, \boldsymbol{\theta}, L) = Q^{-1}(\tau_2)\sqrt{\sum_{i=1}^N \theta_i v_i} + \sum_{i=1}^N \theta_i h_i, \tag{12}$$

where $Q(\cdot)$ is the cumulative distribution function of the normal distribution with zero mean and unit variance. This completes the design of Test 2 of the TFNP method. We finally note that the choice of the individual rates τ_1 and τ_2 is a design issue and in this study, we share the overall false alarm rate τ (specified by the user) equally between individual tests, i.e., $\tau_1 = \tau_2 = 1 - \sqrt{1-\tau}$, unless stated otherwise.

We point out the proposed anomaly detection tests AMCNP and TFNP are appropriate for fast streaming data applications since they require only limited (in the case of AMCNP) or even negligible (in the case of TFNP) computational cost. One can sequentially process data at large scales by using the online implementations of our tests and detect anomalies with real time false alarm rate controllability in a truly online manner. Moreover, no parameter tuning is required to match the desired false alarm rate even when the source statistics change; instead, the proposed tests can actually be implemented adaptive to the possible non-stationarities. For the details of the online implementation of -for instance- a reduced version of the test TFNP as well as the adaptivity to the non-stationarity, please refer to the study [14]. Nevertheless, we briefly note here that: i) The signal features $\boldsymbol{\theta}$, k_i^+ and k_i^- can all be computed online since the computation is based on a simple incremental counting of the transitions as the data is streamed. ii) Once these features are computed online, then the unknown model parameters can be estimated through basic operations, i.e., $\hat{\lambda}_{i(i+1)} = \frac{k_i^+}{\theta_i}$. iii) Once the model parameters are estimated, then the log-likelihood z can be computed and updated online, i.e., $z^{(n+1)} = z^{(n)} + \log\frac{\hat{\pi}_{w_2}}{\hat{\pi}_{w_1}}\frac{\hat{\lambda}_{w_{L+1}}}{\hat{\lambda}_{w_1}}$, if $w_1 = w_2$ and $w_L = w_{L+1}$. Finally, iv) the computation to obtain the desired threshold of TFNP is again a simple function evaluation, cf. Eqs. 11 and 12. For the details, please refer to the sequential data processing framework of [14].

5 Experiments

In this section, we demonstrate the anomaly detection power and the false alarm rate controllability capabilities of the proposed tests AMCNP and TFNP in comparison to the

baseline MCNP. We also present the computational running times of our techniques.

For this purpose, we design a set of experiments, where we use a $N = 4$ state birth-death type DTMC with a specific set of model parameters (these parameters are randomly chosen), i.e.,

$$\mathbf{R} = \begin{bmatrix} \lambda_{10} & \lambda_{12} & \lambda_{11} \\ \lambda_{21} & \lambda_{23} & \lambda_{22} \\ \lambda_{32} & \lambda_{34} & \lambda_{33} \\ \lambda_{43} & \lambda_{45} & \lambda_{44} \end{bmatrix} = \begin{bmatrix} 0 & 0.78 & 0.22 \\ 0.27 & 0.35 & 0.38 \\ 0.63 & 0.31 & 0.06 \\ 0.54 & 0 & 0.46 \end{bmatrix},$$

to generate a nominal set of 10^5 sequences of length $L = 100$. Note that $\lambda_{10} = \lambda_{45} = 0$ is used for convention. We also generate 10^5 examples of anomalous sequences similarly from a $N = 4$ state birth-death type DTMC; however, we randomly change the model parameters for each of these anomalous sequences. Therefore, the anomalous behavior is simulated in this study as a change in the model parameters. For every nominal and anomalous sequence (in total 2×10^5 sequences are tested), we apply the following tests at various desired false alarm rates $\tau \in \{0.01, 0.1, 0.2, \cdots, 0.9, 0.99\}$: i) TFNP with $\tau_1 = \tau_2 = 1 - \sqrt{1-\tau}$, ii) TFNP-A with $\tau_1 = \tau$ and $\tau_2 = 0$, iii) TFNP-B with $\tau_1 = 0$ and $\tau_2 = \tau$, iv) AMCNP and finally v) MCNP. Here, we perform the tests TFNP-A and TFNP-B in order to clearly demonstrate the efficacy of the hierarchical combination of TFNP-A and TFNP-B by TFNP. MCNP is used as the baseline which is the direct application of the NP test via extensive Monte Carlo simulations. Note that MCNP is capable of precisely matching the desired false alarm rate with these simulations. On the other hand, the false alarm controllability by our methods TFNP and AMCNP is nearly achieved based on our log-likelihood approximation; however, in a computationally highly scalable and efficient manner.

In Fig. 3, we present the Receiver Operating Characteristics (ROC) curves for these compared tests. We observe that the proposed test TFNP significantly outperforms the baseline MCNP since the anomalies are simulated in this paper as a change in the source statistics. If the anomalies were uniformly distributed, then it is already known that MCNP would be the optimal, i.e., the most powerful, by definition without a need for demonstration. On the other hand, MCNP is well approximated by our method AMCNP as illustrated in Fig. 3, hence, AMCNP might be a better option (compared to TFNP) when the anomalies are uniformly distributed. Note that the method TFNP first applies the test TFNP-A at the rate τ_1, and then applies the test TFNP-B at the rate τ_2. This combination, i.e., the successive application in the TFNP, is clearly shown to outperform the individual tests since TFNP performs significantly better than TFNP-A and TFNP-B, cf. Fig. 3. In terms of the false alarm rate controllability, we mention here a couple of examples in

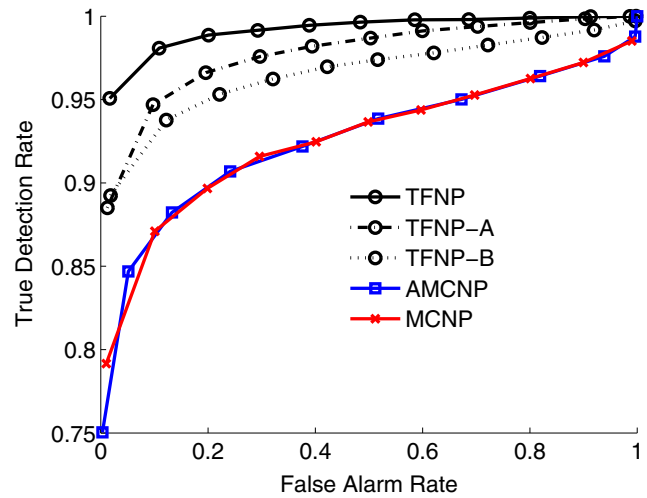

Figure 3 The ROC curves are presented for the compared tests TFNP, TFNP-A, TFNP-B, AMCNP and MCNP.

the case of TFNP. When the desired rate is applied from $\tau = \{0.01, 0.1, 0.2, \cdots, 0.9, 0.99\}$, TFNP impressively achieves these desired rates as $\hat{\tau} = \{0.016, 0.109, 0.200, 0.292, 0.389, 0.484, 0.585, 0.686, 0.799, 0.913, 0.997\}$, cf. Fig. 3. Finally, the running times of the optimized MATLAB codes on a standard work station for the Monte Carlo simulations of the tests AMCNP and MCNP are 74.9 and 3.7 seconds, respectively. Namely, AMCNP is $\sim 25\times$ faster than MCNP. On the other hand, TFNP operates without such Monte Carlo simulations taking almost negligible time compared to AMCNP or MCNP. Therefore, the proposed techniques are computationally highly efficient and scalable and can be readily applied, by using the sequential processing framework of [14], to fast streaming data in the contemporary online applications.

6 Conclusion

Two Neyman-Pearson (NP) tests are introduced for anomaly detection, which are both based on the log-likelihood density approximation that we derive for observations from the birth-death type DTMCs. The first test "AMCNP" uses this approximation to perform a "light" set of Monte Carlo simulations and "nearly" matches the desired false alarm rate. The second test "TFNP" applies a two fold NP tests against certain statistics and only requires simple function evaluations to "nearly" match the desired rate, i.e., to find the corresponding approximate density quantile, without Monte Carlo simulations. AMCNP is appropriate for detecting uniformly distributed, i.e., arbitrary, anomalies and operates $\sim 25\times$ faster than the baseline NP, which requires "heavy" Monte Carlo simulations to "precisely" match the desired rate. TFNP operates incomparably faster than even AMCNP

without simulations since its computational complexity is relatively negligible; and it is appropriate for detecting -not uniformly distributed but- the anomalies that are still drawn from birth-death type chains with different parameters (not with the nominal ones). Therefore, the introduced tests can be used to process data in large scales with real time false alarm rate controllability. Moreover, our algorithms do not require parameter tuning even under strong non-stationarity. Through several numerical examples in our experiments, we show that the proposed algorithms outperform the baseline NP in terms of the detection power, while nearly achieving the desired rate in a computationally highly scalable manner.

Acknowledgments This work was supported in part by The Scientific and Technological Research Council of Turkey (TUBITAK) under Contract 113E517 and in part by Turk Telekom Inc.

References

1. Chandola V., Mithal V., & Kumar V. (2008). Comparative evaluation of anomaly detection techniques for sequence data. In *International conference on data mining* (pp. 743–748).
2. Chandola, V., Banerjee, A., & Kumar, V. (2009). Anomaly detection: a survey. *ACM Computing Surveys (CSUR)*, *41*(3), 15.
3. Chandola, V., Banerjee, A., & Kumar, V. (2012). Anomaly detection for discrete sequences: a survey. *IEEE Transactions on Knowledge and Data Engineering*, *24*(5), 823–839.
4. Filippone, M., & Sanguinetti, G. (2011). A perturbative approach to novelty detection in autoregressive models. *IEEE Transactions on Signal Processing*, *59*(3), 1027–1036.
5. Gupta, M., Gao, J., Aggarwal, C., & Han, J. (2014). Outlier detection for temporal data: a survey. *IEEE Transactions on Knowledge and Data Engineering*, *26*(9), 2250–2267.
6. Karlin, S. (2014). *A First Course in Stochastic Processes*. Academic Press.
7. Keogh, E., Lin, J., Lee, S., & Van Herle, H. (2007). Finding the most unusual time series subsequence: algorithms and applications. *Knowledge and Information Systems*, *11*(1), 1–27.
8. Lehtomaki, J., Vartiainen, J., Juntti, M., & Saarnisaari, H. (2007). Cfar outlier detection with forward methods. *IEEE Transactions on Signal Processing*, *55*(9), 4702–4706.
9. Marceau, C. (2001). Characterizing the behavior of a program using multiple-length n-grams. In *Proceedings of the 2000 workshop on New security paradigms* (pp. 101–110). ACM.
10. Michael C, & Ghosh A (2000). Two state-based approaches to program-based anomaly detection. In *Annual conference on computer security applications* (pp. 21–30).
11. Morgan, D. (2007). On level-crossing excursions of gaussian low-pass random processes. *IEEE Transactions on Signal Processing*, *55*(7), 3623–3632.
12. Moser, B., & Natschlager, T. (2014). On stability of distance measures for event sequences induced by level-crossing sampling. *IEEE Transactions on Signal Processing*, *62*(8), 1987–1999.
13. Ozkan, H., Akman, A., & Kozat, S. (2014). A novel and robust parameter training approach for hmms under noisy and partial access to states. *Signal Processing*, *94*, 490–497.
14. Ozkan, H., Ozkan, F., Delibalta, I., & Kozat, S. (2015). Online anomaly detection with constant false alarm rate. In *IEEE 25th international workshop on machine learning for signal processing (MLSP)* (pp. 1–6).
15. Ozkan, H., Pelvan, O., & Kozat, S. (2015). Data imputation through the identification of local anomalies. *IEEE Transactions on Neural Networks and Learning Systems*, *26*(10), 2381–2395.
16. Poor, V. (1994). *An introduction to signal detection and estimation*. Springer Science & Business Media.
17. Rajasegarar, S., Leckie, C., & Palaniswami, M. (2008). Anomaly detection in wireless sensor networks. *IEEE Wireless Communications*, *15*(4), 34–40.
18. Ramaswamy, S., Rastogi, R., & Shim, K. (2000). Efficient algorithms for mining outliers from large data sets. *ACM SIGMOD Record*, *29*(2), 427–438.
19. Saligrama, V., Konrad, J., & Jodoin, P. (2010). Video anomaly identification. *IEEE Signal Processing Magazine*, *27*(5), 18–33.
20. Scott, C., & Nowak, R. (2006). Learning minimum volume sets. *The Journal of Machine Learning Research*, *7*, 665–704.
21. Wang, H., Tang, M., Park, Y., & Priebe, C. (2014). Locality statistics for anomaly detection in time series of graphs. *IEEE Transactions on Signal Processing*, *62*(3), 703–717.
22. Xue, M., & Roy, S. (2011). Spectral and graph-theoretic bounds on steady-state-probability estimation performance for an ergodic markov chain. In *American control conference* (pp. 2399–2404).
23. Ye, N., et al. (2000). *A markov chain model of temporal behavior for anomaly detection* (Vol. 166, p. 169).
24. Zhang, K., Hutter, M., & Jin, H. (2009). A new local distance-based outlier detection approach for scattered real-world data. In *Advances in knowledge discovery and data mining* (pp. 813–822). Springer.
25. Zhao, M., & Saligrama, V. (2009). Anomaly detection with score functions based on nearest neighbor graphs. In *Advances in neural information processing systems* (pp. 2250–2258).
26. Zivkovic, Z. (2004). Improved adaptive gaussian mixture model for background subtraction. *IEEE 17th International Conference on Pattern Recognition (ICPR)*, *2*, 28–31.

Huseyin Ozkan is -as a postdoctoral scholar- currently with the Department of Brain and Cognitive Sciences at Massachusetts Institute of Technology (MIT), Cambridge, MA, USA. He received his B.Sc. degrees in Electrical and Electronics Engineering, and Mathematics from Bogazici University, Istanbul, Turkey, in 2007. He received his M.Sc. degree in Electrical Engineering from Boston University, MA, USA, in 2010; and his Ph.D. degree in Electrical and Electronics Engineering from Bilkent University, Ankara, Turkey, in 2015. Before joining CSAIL at MIT, he worked at Aselsan Inc., Ankara, Turkey, as a research scientist, where he conducted computer vision research for large area surveillance; and also focused on anomaly detection and recommendation problems. 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. He 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). His research interests include statistical learning, pattern recognition, computer vision and statistical signal processing.

Fatih Ozkan received his B.Sc. degree in Computer Engineering from Cukurova University, Adana, Turkey, in 2012. He is currently working towards his M.Sc. degree in the Department of Information Systems at 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.

Ibrahim Delibalta received his B.S. in EE from Bogazici University, Istanbul in 1993 and M.S. in EE from University of Southern California in 1995. He worked in Silicon Valley from 1995 to 2010 at Intel, Silicon Graphics and Cisco Systems, respectively, in the field of high performance microprocessor and network processor chip design. He has been working in the Turk Telekom Group since 2011 at various positions, currently leading the Emerging Services and R&D team. He is currently pursuing a Ph.D. in Design, Technology and Society, an interdisciplinary program at Koc University, Istanbul, involving social sciences and machine learning.

Suleyman S. Kozat received his B.S. degree in Electrical and Electronics Engineering from Bilkent University, Ankara, Turkey. He received the M.S. and Ph.D. degrees in Electrical Engineering from University of Illinois at Urbana Champaign, IL, USA, in 2001 and 2004, respectively. After graduation, Dr. Kozat 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, Dr. Kozat is an Assistant Professor at 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 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 Career Award by the Scientific Research Council of Turkey, 2009.