Received December 6, 2019, accepted January 15, 2020, date of publication February 10, 2020, date of current version February 18, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.2972640

Real-Time Algorithms for the Detection of Changes in the Variance of Video Content Popularity

SOTIRIS SKAPERAS[1], (Student Member, IEEE), LEFTERIS MAMATAS[1], (Member, IEEE), AND ARSENIA CHORTI[2], (Member, IEEE)

[1]Department of Applied Informatics, University of Macedonia, 546 36 Thessaloniki, Greece
[2]ETIS/Université Paris Seine, Université Cergy-Pointoise, ENSEA, CNRS, 95000 Cergy, France

Corresponding author: Sotiris Skaperas (sotskap@uom.edu.gr)

This work was supported in part by the EU's Horizon 2020 Research and Innovation Programme through the 4th open call scheme of the FED4FIRE+ under Grant 732638, in part by the EU-BRA Horizon 2020 NECOS Project under Grant 777067, in part by the Initiative d'Excellence Paris//Seine project eNiGMA, and in part by the ELIOT project under Grant ANR-18-CE40-0030/FAPESP 2018/12579-7.

ABSTRACT As video content is responsible for more than 70% of the global IP traffic, related resource allocation approaches, e.g., using content caching, become increasingly important. In this context, to avoid under-provisioning, it is important to rapidly detect and respond to changes in content popularity dynamics, including volatility, i.e., changes in the second order moment of the underlying process. In this paper, we focus on the early identification of changes in the variance of video content popularity, which we address as a statistical change point (CP) detection problem. Unlike changes in the mean that can be well captured by non-parametric statistical approaches, to address this more demanding problem, we construct a hypothesis test that uses in the test statistic both parametric and non-parametric approaches. In the context of parametric models, we consider linear, in the form of autoregressive moving average (ARMA), and, nonlinear, in the form of generalized autoregressive conditional heteroskedasticity (GARCH) processes. We propose an integrated algorithm that combines off-line and on-line CP schemes, with the off-line scheme used as a training (learning) phase. The algorithm is first assessed over synthetic data; our analysis demonstrates that non parametric and GARCH model based approaches can better generalize and are better suited for content views time series with unknown statistics. Finally, the non-parametric and the GARCH based variations of our proposed integrated algorithm are applied on real YouTube video content views time series, to illustrate the performance of the proposed approach of volatility change detection.

INDEX TERMS Content popularity dynamics detection, change point analysis, variance change detection, volatility detection.

I. INTRODUCTION

Understanding the popularity characteristics of online content and predicting the future popularity of individual videos are of great importance. They have direct implications in various contexts [1], such as service design, advertisement planning, network management [2], and so on. As an example, an efficient content caching scheme should be popularity-driven [3], meaning that it should incorporate the future popularity of content into the caching decision making. In this framework, novel cache replacement methods that are "popularity-driven" have recently appeared, e.g., the algorithms proposed in [4], based on learning the popularity of content and using it to determine which content should be retained and which should be evicted from the cache. Other important applications include content delivery networks (CDNs) in which "analytics-as-a-service" approaches are employed and information centric networks (ICNs) with emphasis on the Internet of things (IoT) [5].

Higher order moments of the underlying random process are unarguably important for the efficient statistical characterization of content popularity; in particular, "volatility" plays a central role in capturing the underlying dynamics of content views. As an example, in caching applications, it has been established in [6] that a major factor greatly impacting

The associate editor coordinating the review of this manuscript and approving it for publication was Jose Saldana.

efficiency is related to demand volatility; this reflects the fact that files might not be constantly requested following a stationary model, but rather, only be requested once or twice and subsequently exhibit vanishing demand in time (e.g., volatility in YouTube content). Based on these findings, an efficient strategy for resource provisioning should in principle consider not only conditional mean demands but also demand fluctuations, thus avoiding under-provisioning or over-provisioning.

To analyze the underlying statistics of content views data, the latter are typically represented as a time series. Time series data are sequences of measurements over time, describing the behavior of systems. The behavior can change over time due to external events and/or internal systematic changes in dynamics/distribution. Success in revealing such patterns can be translated to the ability to respond rapidly to these changes. In this direction, there has recently been a surge of research in the area of content popularity prediction using artificial intelligence (AI) [7]. In this context, machine learning based methods (e.g., deep learning) need effective feature mining and a huge mass of labeled examples to provide successful performance [8], [9]. In applications in which *real time* content popularity monitoring is required this might become a challenge. As an example, in [10] the authors propose an *off-line* deep learning approach to detect popularity that is subsequently integrated into the on-line caching policy in fog radio applications; however, whenever there is an important change in the underlying dynamics of content popularity, it follows that a new off-line training might be required to run the algorithm properly.

In this work, we alternatively turn our attention to lightweight *statistical* procedures that fall in the general context of AI (instead of deep learning specifically), in order to operate in an on-line manner (real-time) and to keep the size of the required set of historical data as small as possible. Our proposed algorithm is autonomous, in the sense that all its parameters are determined without manual intervention during a training period; furthermore, the training period is limited to only a few hundred data points (instead of thousands or millions as is typical in deep learning).

Importantly, instead of attempting to *predict* the evolution of content popularity, in this work we rather focus on *detecting* changes in its underlying statistics, and doing so in real-time. To this end, we propose the use of on-line change point (CP) analysis; to complement our work [11], [12] that focused on the identification of changes in the mean of a time series, here, we alternatively investigate the performance of corresponding on-line algorithms to identify changes in the variance of a time series using CP analysis.

In general, CP methods are either off-line or on-line. Off-line algorithms operate retrospectively and identify CPs in a historical dataset, a thorough study can be found in [13]. On-line algorithms [14] monitor in real time a data sequence and aim to detect CPs as soon as they occur. In this work, we propose an efficient combination of an off-line and various on-line procedures for the detection of changes in the second order statistics of video content popularity, as soon as they occur (real-time). The proposed detector is built upon our earlier proposal for a real-time CP detector of mean changes in data series, that we applied to monitor the average number of video content [11], [12]. Albeit, the monitoring of changes in the variance of a time series is much more challenging.

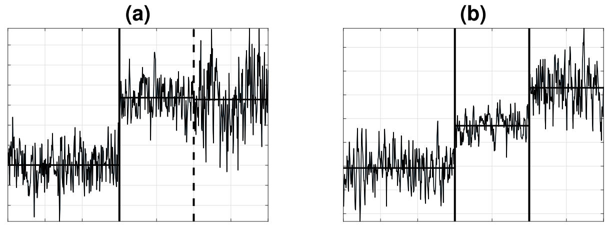

FIGURE 1. Simulated time series with CPs in the mean (solid line) and the variance (dashed line) for (a) separated and (b) simultaneous changes in the mean/variance. Horizontal lines illustrate the mean value.

To further illustrate our motivation behind this work, we note that an overall approach considering both mean and variance changes allows for a more efficient handling of content popularity changes as highlighted in Fig. 1. For example, Fig. 1(a) depicts that a crucial popularity change may affect only the variance parameter, in the specific example at the third segment of the time series. On the other hand, Fig. 1(b), depicts that in the case of a simultaneous change in the mean and the variance, e.g., in the second segment of the time series, the latter is critical to estimate the actual impact of this change. Monitoring the variance may also be used as a measure of uncertainty, determining the degree of fluctuation of popularity around its expectation; for instance, compare the behaviour of the time series in Fig. 1(b) after the first and the second CP (second and third segments of the data series, respectively).

To identify changes in the variance, a more elaborate test statistic is employed in the present study. With respect to [11], [12], we further introduce novel on-line tracking mechanisms based on autoregressive moving average (ARMA) and generalized autoregressive conditional heteroskedasticity (GARCH) models. The most important novel aspects of this paper are listed below:

- We show that variance CP detection is important in the context of content popularity.
- We introduce a relevant on-line detection algorithm, enhanced by the following two mechanisms: (a) an offline CP detection over training data for the estimation of the on-line test parameters; and (b) identification of the change magnitude in the pro- and post-change variance structure.
- Our algorithm supports three alternative on-line tests for content popularity detection – based on ARMA and GARCH models as well as a non-parametric approach – covering a wide-range of time series characteristics.
- We performed experiments both on synthetic and real time series datasets. Our results show that:

(i) the GARCH and the non-parametric approaches perform better when the time series does not follow a linear model; (ii) overall, these approaches can generalize better with respect to the true alarm rates; and (iii) the non parametric approach can identify CPs more rapidly.

In future work we intend to expand the algorithm to include additional dimensions that can be volatility indicators, such as the number of likes, viewer comments, content size, as well as network parameters such as the utilization of servers, in order to enhance the agility of the volatility estimation of the so called "content workload" as a whole. We will also investigate the algorithm's scalability properties, theoretically and experimentally, i.e., identify the number of videos that can be analyzed in parallel.

The rest of this contribution is structured as follows: In Section II, background concepts and high level properties of the proposed integrated algorithm are discussed. In Section III, the offline training is presented in detail, while Section IV presents three different approaches for the construction of the online test statistic. The integrated algorithms are assessed on synthetic data in Section V and applied to real YouTube content view data in Section VI. Conclusions and discussion on future enhancements are included in Section VII.

II. BACKGROUND CONCEPTS AND INTEGRATED ALGORITHM

A. CHANGE POINT ANALYSIS

Change point (CP) detection refers to the problem of identifying data structures that do not correspond to the anticipated "normal" behavior. We note that, to the best of our knowledge, this is the first work in the literature proposing an automated mechanism for the detection of volatility changes in a time series in the context of content popularity detection.

The theory of CP analysis is typically pertinent to anomaly detection. In the domain of networking in particular, the theory of CP detection has played an instrumental role in the modelling of network traffic monitoring represented through time series [15] and network anomaly/intrusion detection [16]; for a comprehensive review the interested reader may refer to [17]. In this framework, CP detection techniques [18] are used for the identification of: (i) point anomalies and outliers, i.e., data points deviating distinctively from the bulk of collected data; (ii) pattern anomalies, i.e., groups of data points that are collectively anomalous with respect to historical data; and, (iii) CP anomalies due to changes in the time series's statistical structure (in the mean/variance and in general in the underlying distribution). In this work, we focus on the detection of CP anomalies and consider the other two categories as disturbances. The reasoning behind this choice is that, on one hand, a resource allocation scheduler should be insensitive to instantaneous/very short-term changes in resource demand (e.g., represented as outliers in the content demand), but, on the other hand, should be highly responsive to changes in the underlying statistics of the demand.

B. PARAMETRIC AND NON PARAMETRIC CP DETECTION ALGORITHMS

Statistical based approaches are categorized as parametric [19] and non-parametric [14]. Non-parametric methods do not make use of a particular time series model fit and apply directly the observed data to the monitoring procedures. In this context, CUSUM based methods are non-parametric by design. For example, the authors in [20] provide a CUSUM stopping rule with application in computer vision problems. A CUSUM approach for CP detection on observations with an unknown distribution before and after a change, has been recently developed in [21]. Furthermore, an algorithm based on the Shiryaev-Roberts procedure was proposed in [22], to detect anomalies in computer network traffic.

On the other hand, parametric methods utilize as inputs values obtained from a specific model that has been fit to the original data (instead of using the original data set directly). As an example, Kalman filtering is combined with several CP methods in [23]. In [24], traffic flows are modeled using Markov chains and an anomaly detection mechanism based on the generalized likelihood ratio test (LRT) algorithm. Further examples assuming specific distribution for the data include [25], in which a bivariate sequential generalized LRT algorithm was proposed, assuming that the packet rate and the packet size follow a Poisson and a normal distribution, respectively. Other, non residual methods, include estimates' detectors based on the differences between the estimated model parameters (see [13], [26]), or based on the quasi-likelihood scores estimators of the parameters of a GARCH process [27].

C. VIDEO CONTENT POPULARITY PREDICTION VS DETECTION

The prediction of video content popularity characteristics and dynamics [28], as well as models to predict popularity evolution, e.g., [29] and [30], is a well studied topic in the literature. Among others, in [31], the authors perform a detailed analysis to characterize the YouTube traffic within a campus network and conclude that in this scenario the content popularity can be well approximated by the Zipf distribution. A comprehensive survey on video traffic models can be found in [32]. Overall, several methods have been proposed in this context, including time series models, regression models [33]–[35] and machine learning (deep neural networks) techniques [36], [37].

Focusing on time series modelling in particular, linear, non linear and hybrid models have invariably been proposed. In early works, linear time series models have been used, e.g., the authors in [38] introduce an ARMA(7, 7) model to describe and predict the daily views of individual videos. Alternatively, in [39], by taking into consideration seasonality, an autoregressive integrated moving average (ARIMA) model is used to forecast the popularity of online content. Other approaches include fractional ARIMA (FARIMA) models, that capture both short-range dependence (SRD) and long-range dependence (LRD) statistical properties [40].

Recently, non linear models have further been proposed to take into account the conditional heteroskedasticity and the conditional volatility of the data series (seen as a stochastic process). In these cases, GARCH models are involved. For example, in the comparative study [41], the authors showed that a hybrid ARIMA/GARCH model was superior to FARIMA and wavelet neural network models, while in [42], a similar hybrid FARIMA/GARCH approach was also introduced. In essence, the existing hybrid models consider the second order characteristics of a time series as a supplementary element to further improve the forecasting or estimation of the content popularity. More precisely, these solutions assume conditional heteroskedasticity for the errors of the ARMA or FARIMA model. An exception can be found in [43], where a video demand predictor forecasts the volatility and correlation of the streaming traffic associated with different videos, based on multivariate GARCH models.

On the other hand, the problem of detecting (i.e., estimating), non-parametrically and in real time, CPs on content popularity sequences, has not been adequately investigated yet. Among of only a handful of related studies, in our previous works [11], [12], [44] we proposed and implemented a real-time, non-parametric and low-complexity video content popularity CP detector (as opposed to predictor) for changes in the mean value of video content popularity. In the present contribution, in contrast to [11], [12], we introduce an innovative online algorithm for the detection of CPs in the second order statistics of content popularity data. We also present an enlarged statistical framework, that includes parametric as well as non-parametric detectors.

Our algorithm can be used as a "stand alone" mechanism, but may also be a helpful complementary tool for prediction approaches. With respect to the latter, it can be employed in validating whether assumptions made by a prediction model are still reasonably satisfied, or, whether the prediction model/procedure needs adjustment. Since, data are often influenced by a multitude of external factors, stationarity assumptions cannot be guaranteed over the whole monitoring period, especially for long time ranges.

D. OVERVIEW OF THE PROPOSED INTEGRATED ALGORITHM

We summarize in Fig. 2 the overall algorithm as a flow diagram that links an off-line (training) and an on-line phase, as well as their individual components. Without loss of generality we assume an arbitrary time instance m_s as the starting point of a monitoring period. Then, the off-line analysis is applied to the historical (training) data until $t = m_s$, resulting in the division of the data sequence in stable subsequences. The last subsequence is the training sample representing the initial sample of the on-line phase. During the training stage, if a parametric approach is chosen, we estimate the model parameters (e.g., ARMA or GARCH) and any other necessary statistical characteristics that describe the last stable subsequence's (time series) behavior. We note that without having first obtained a statistically robust division of the training sample into stable subsequences, the estimation of a model's parameters could be seriously impacted.

Next, an on-line detector is implemented for a monitoring period $t = m_{s+1}, \ldots, m_{s+l}$. If a CP is detected at cp^*_{on}, the CP magnitude on the data structure is evaluated. The new starting point for the subsequent monitoring window is then set to $m'_s = cp^*_{on} + d$, where d is a constant specifying a period assuming no change. Alternatively, if no change is detected after l instances, the procedure restarts automatically from the time point $m'_s = m_{s+l}$. The reasons behind this choice are twofold. First, to keep the algorithm running over a window of size at most l, in order to keep the computational complexity low (lightweight), as opposed to allowing increasing window sizes. Second, to facilitate the fast responsiveness of the algorithm, as will be demonstrated through numerical examples in Section V.

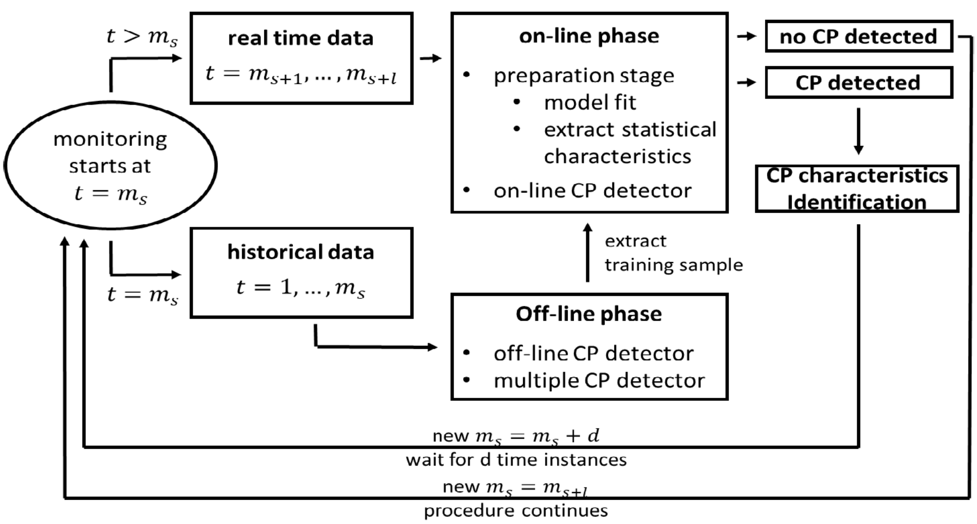

FIGURE 2. Flow diagram of the real-time variance CP detector for content views data.

III. OFF-LINE PHASE

In this Section, the training phase of the algorithm is discussed and the fundamental components of the off-line scheme are presented. We choose a retrospective CP scheme to ascertain that the on-line phase is indeed carried out on homogeneous data. We note that standard off-line CP schemes can only detect a single CP. To address the issue of detection of multiple CPs, we modify the basic scheme with a novel time series segmentation heuristic, that belongs to the family of binary segmentation algorithms, similarly to [11], [12].

Let $\{X_n : n \in \mathbb{N}\}$ be a time series representing the content views, for a specific video. Since we are interested only in the variance fluctuation of the underlying random value (r.v.), we assume a constant, over time, expected value $E(X_i)$, where $E(\cdot)$ denotes expectation. The stability of the mean value can be ensured by a data transformation, such as taking the first differences , $\Delta_n = X_n - X_{n-1}$, thus rendering $E(X_i) = 0$.

Considering the training phase, we have to check if the variance structure remains stable over the whole training period N. Consequently we study the null hypothesis,

$$H_0 : \sigma_1^2 = \cdots = \sigma_N^2, \tag{1}$$

where $\sigma_n^2 = \mathrm{Var}(X_n) = E(X_n^2)$, given that we have modified the time series so that $E(X_n) = 0$. The (general) alternative hypothesis is designed to allow the existence of multiple changes $l_i \in \{1, \ldots, N\}$, $i = 1, \ldots, r$, where r is the multitude of changes,

$$\begin{aligned} H_1 :\ & \sigma_1^2 = \ldots = \sigma_{l_1}^2 \neq \sigma_{l_1+1}^2 = \ldots = \sigma_{l_2}^2 \neq \ldots \\ & \cdots \neq \sigma_{l_{\tau-1}+1}^2 = \ldots = \sigma_{l_\tau}^2 \neq \sigma_{l_\tau+1}^2 = \ldots = \sigma_N^2. \end{aligned} \tag{2}$$

We develop a CP detector that only requires very general sufficient assumptions to be satisfied by the time series of content views. More specifically, we followed the work in [45] in which the authors introduce a non-parametric test statistic that requires only that the time series $\{X_n : n \in \mathbb{N}\}$ can be approximated, with a distance measure, by an s-dependent r.v. This assumption assures that time series needs not be s-dependent itself. We also note that several popular weak dependent time series models for the description of video views satisfy the above assumption, e.g., ARMA or GARCH models. The exact form of the procedure is given in the quadratic scheme,

$$TS_N^{off} = \frac{1}{N} S_n^T \hat{\Omega}_N^{-1} S_n, \tag{3}$$

with $(\cdot)^T$ denoting transposition, and, it converges in distribution asymptotically to,

$$\int_0^1 B^2(n)dn, \quad (N \to \infty), \tag{4}$$

where $(B(n) : n \in [0, 1])$ are independent standard Brownian bridges. (4) can be used to derive the critical values (cv^{off}) of the test statistic TS_N^{off} by Monte Carlo simulations that approximate the paths of the Brownian bridge on a fine grid. As an example, using this approach, the crossing boundaries of (4) for alarm rates of 5% and 1% can be found to be 1.8 and 2.6, respectively.

The detector S_n is a variation of the squared CUSUM method,

$$S_n = \frac{1}{\sqrt{N}} \left(\sum_{i=1}^{n} \mathrm{vech}[\tilde{X}_i \tilde{X}_i^T] - \frac{n}{N} \sum_{i=1}^{N} \mathrm{vech}[\tilde{X}_i \tilde{X}_i^T] \right), \tag{5}$$

where the vech(.) operator denotes the half-vectorization of a matrix (as the covariance matrix is symmetric, half-vectorization contains all the strictly necessary information) and $\tilde{X}_i = X_i - \overline{X}_N$, with $\overline{X}_N = \frac{1}{N} \sum_{j=1}^{N} X_j$ the sample average.

Since the procedure (3) is non-parametric, the dependence between the observations enters only in the form of the long-run covariance Ω_N, expressed as

$$\Omega_N = \sum_{i=1}^{N} \mathrm{Cov}(\mathrm{vech}[X_0 X_0^T], \mathrm{vech}[X_i X_i^T]) \tag{6}$$

To build a consistent estimator of Ω_N, denoted by $\hat{\Omega}_N$, various different approaches exist. This estimation problem is well studied and we focus on the kernel based approach through the use of Newey-West estimator (see [46]),

$$\widehat{\Omega}_N = \widehat{\Sigma}_0 + \sum_{w=1}^{W} k_{BT}\left(\frac{w}{W+1}\right)\left(\widehat{\Sigma}_w + \widehat{\Sigma}_w^T\right), \tag{7}$$

where $k_{BT}(.)$ corresponds to the Bartlett weight,

$$k_{BT}(x) = \begin{cases} 1 - |x|, & \text{for } |x| \leqslant 1 \\ 0, & \text{otherwise,} \end{cases} \tag{8}$$

and $\widehat{\Sigma}_w$ denotes the empirical auto-covariance matrix for lag w,

$$\widehat{\Sigma}_w = \frac{1}{N} \sum_{n=w+1}^{N} \left(X_n - \overline{X}\right)\left(X_{n-w} - \overline{X}\right)^T. \tag{9}$$

Following common practice in literature we chose $W = \log_{10}(N)$. To summarize, the existence of a CP is announced if $TS_N^{off} > cv_V^{off}$ and the estimated time of change is,

$$cp_{off}^* = \frac{1}{N} \underset{1 \leqslant n \leqslant N}{\mathrm{argmax}}\, TS_N^{off}. \tag{10}$$

Finally, to face the potential of detecting multiple CPs on the historical data set, we have integrated an extended version of the binary segmentation (BS) algorithm, proposed in [11], to the original test TS_N^{off}. The algorithm combines the standard BS and the iterative cumulative sum of squares (ICSS) [47] methods and operates briefly as follows: First, a single CP is searched in the historical sample. In case of no change, the procedure stops and H_0 is accepted. Otherwise, the detected CP is used to divide the time series into two time series in which new searches are performed. The procedure is iterated, until no more CPs are detected. In the last step, we consider the CPs estimated previously in pairs and check if H_0 is still rejected in the segment delimited by each pair. If not, the CPs that fall in the particular segment are eliminated.

IV. ON-LINE METHODS

In this Section we present three alternative on-line approaches and discuss jointly for each one the preparation stage and the corresponding on-line CP detector. The on-line phase is based on the assumption of an homogeneous data sequence of length $m \in \mathbb{N}^+$, determined by the off-line phase, for which,

$$\sigma_1^2 = \cdots = \sigma_m^2. \tag{11}$$

Our aim is to test if (11) holds as new observations become available in a time real framework. Hence, the statistical problem is formulated as the following hypothesis test,

$$\begin{aligned} &H_0 : \sigma_1^2 = \cdots = \sigma_m^2 = \sigma_{m+1}^2 = \cdots, \\ &H_1 : \sigma_{m+1}^2 = \cdots = \sigma_{m+l-1}^2 \neq \sigma_{m+l}^2 = \sigma_{m+l+1}^2 \cdots, \\ &\quad m, l \in \mathbb{N}^+. \end{aligned} \tag{12}$$

In general, any on-line CP method can be described as a stopping time procedure with stopping time $\tau(m)$,

$$\tau(m) = \min\{l \in \mathbb{N} : TS^{on}(m, l) \geq b\}. \tag{13}$$

The value of the test statistic $TS^{on}(m, l)$ is calculated online for every l in the monitoring period. The rule stops, and a change is announced, if the test statistic exceeds the boundary function $b = cv^{on}g$. The critical value cv^{on} is derived from the asymptotic behavior of the detector TS^{on}/g under the null hypothesis, for which $Pr\,(\tau(m) < \infty) = \alpha$, $\alpha \in (0, 1)$ the significance level. We note that γ, $\gamma \in (0, \frac{1}{2}]$ is a sensitivity parameter; the larger the value of γ, the smaller the value of b, which leads to a quicker detection of a potential CP, at the cost of an increase in the false alarm rate.

Below, we consider three on-line CP approaches, based on the general assumptions for the underlying process: i) a non-parametric approach based on [48], denoted by NP; ii) a linear time series (ARMA) approach as in [49], denoted by L; and, iii) a nonlinear time series (GARCH) approach like in [50], denoted by NL. The quantities $\{TS^{on}, b, cv^{on}, g\}$ will be indexed accordingly.

A. NON-PARAMETRIC (NP) APPROACH

Non-parametric approaches work directly with the observed data and are ideal for datasets with a high degree of model fitting ambiguity. In this framework, in the preparation phase we only compute a particular form of the long-run estimator, avoiding the difficulties related to the estimation of a parametric model.

The proposed procedure is applied under the assumption that the observations $\{X_n : n \in \mathbb{Z}\}$ satisfy the generalized dependence concept of L-2 near epoch dependence (see [51]). Since the test is model-independent, the dependence between observations is captured through the long-run function D_n, expressed as

$$D_n := \lim_{n\to\infty} \mathrm{E}\left(\frac{1}{n}A_i A_i^T\right), \tag{14}$$

where $A_i = \sum_{t=1}^{i}\left(X_t^2 - \mathrm{E}\left(X_t^2\right)\right)$. We also assume that D_n is finite under the H_0 hypothesis, which is necessary for the convergence of the asymptotic null behaviour.

As explained above, the long-run factor is computed in the preparation phase, considering the training sample. For its evaluation we choose the kernel estimation method, as in [52]. More specifically,

$$\hat{D}_m = \sum_{i=1}^{u}\sum_{j=1}^{u} k_{BT}\left(\frac{i-j}{r}\right)\hat{V}_i\hat{V}_j^T, \tag{15}$$

is an estimator of D_m, $\hat{V}_t = \frac{1}{\sqrt{m}}\left(X_t^2 - \frac{1}{m}\sum_{i=1}^{m} X_i^2\right)$ and $k_{BT}(\cdot)$ is the Bartlett kernel, already mentioned in (7).

The test statistic is expressed as

$$TS_{NP}^{on}(m, l) = \frac{l}{\sqrt{m}}\hat{D}_m^{-\frac{1}{2}}\left(\sum_{i=m}^{m+l} X_i^2 - \frac{1}{m}\sum_{i=1}^{m} X_i^2.\right) \tag{16}$$

The boundary function $b_{NP} = cv_{NP}^{on}g_{NP}$ is strictly aligned with the chosen size of the monitoring period l normalized to the length of the training period, denoted by $H = l/m$. Then the weight function is expressed as $g_{NP} = \left(1 + \frac{l}{m}\right)\left(\frac{l}{m+l}\right)^{\gamma}$, $\gamma \in [0, 1/2)$ and the critical value is derived from the asymptotic behavior of the stopping rule,

$$\begin{aligned} &\lim_{m\to\infty} Pr\{\tau(m) < \infty\} \\ &\quad = \lim_{m\to\infty} Pr\left\{TS_{NP}^{on} \geq b_{NP}(\alpha)\right\} \\ &\quad = \lim_{m\to\infty} Pr\left\{\frac{TS_{NP}^{on}}{g_{NP}} \geq c_{NP}^{on}(\alpha)\right\} \\ &\quad = Pr\left(\sup_{n\in[0,1]}\left(\frac{H}{1+H}\right)^{\frac{1}{2}-\gamma}\frac{|W(n)|}{n^{\gamma}}\right) = \alpha. \end{aligned} \tag{17}$$

B. LINEAR (L) PARAMETRIC APPROACH USING AN AUTOREGRESSIVE MOVING AVERAGE (ARMA) MODEL

Parametric approaches, monitor the estimated values obtained from a specific model fit to the observed time-series. This is very efficient whenever a parametric model sufficiently describes the dependence structure of the real data. We present two residual based parametric schemes, constructed from the residuals of the model fit to the data, starting with an ARMA model. In the preparation stage, the model residuals are estimated, under the assumption of a homogeneous underlying process. Under H_0, the residuals before and after the beginning of the monitoring should behave similarly. On the other hand, if a CP exists in the monitoring period, the residuals are expected to deviate from those in the training period.

ARMA processes provide linear and parsimonious descriptions of (weakly) stationary processes. A time series $\{X_n : n \in \mathbb{N}\}$ is called an ARMA(p, q) process of orders p and q, if it satisfies the stochastic equation,

$$\phi_n(B)(X_n - \mu_n) = \theta_n(B)\epsilon_n, \quad n \in \mathbb{Z}, \tag{18}$$

where μ_n are mean parameters (usually non stationary), $\phi_n(z) = 1-\phi_{1n}z-\cdots-\phi_{pn}z^p$ and $\theta_n(z) = 1-\theta_{1n}z-\cdots-\theta_{qn}z^q$ are the autoregressive and moving average polynomials of the model respectively, and B the backshift operator. It is also assumed that the ARMA process is causal and invertible, i.e.,

$$\phi_n(z) \neq 0 \text{ and } \theta_n(z) \neq 0, \quad \text{for all } |z| \leq 1. \tag{19}$$

The error terms $\{\epsilon_n : n \in \mathbb{Z}\}$ are a sequence of independent and identically distributed (i.i.d) r.v. with zero mean, $E(\epsilon_1) = 0$ and constant variance, $E(\epsilon_1^2) = \sigma^2$.

The ARMA model in (18) depends on $p+q+2$ parameters, represented by the vector $\beta_n = (\mu_n, \phi_n, \theta_n, \sigma_n^2)$, where $\phi_n = (\phi_{1n}, \cdots, \phi_{pn})$ and $\theta_n = (\theta_{1n}, \cdots, \theta_{qn})$. In the defined training period of size m the parameters of the ARMA model are not time dependent, i.e., they are the same for the observations $X_1, \cdots, X_m$, denoted by β_0 in the following,

$$\beta_0 = (\mu_0, \phi_0, \theta_0, \sigma_0^2). \tag{20}$$

The preparation stage is applied to the training sample for two reasons. Firstly, in order to specify the order (p, q) of the corresponding ARMA model, by selecting the combination that provides the lower value for the Bayes information criterion (BIC),

$$BIC = -2\ln(\hat{L}) + k\ln(n), \tag{21}$$

where $\hat{L}$ is the maximum value of the likelihood function of the model, k is the number of the estimated parameters and n is the sample size. Secondly, in order to estimate the parameters β_0 of the ARMA model through the estimators $\hat{\beta}_0 = (\hat{\mu}_0, \hat{\phi}_0, \hat{\theta}_0, \hat{\sigma}_0^2)$, computed, for example, by the method of maximum likelihood estimation or least squares.

Then, the model residuals are given by

$$\hat{\epsilon}_n = \hat{X}_n - \sum_{i=1}^{p} \hat{\phi}_{i0}\hat{X}_{n-i} - \sum_{i=1}^{q} \hat{\theta}_{i0}\hat{\epsilon}_{n-i}, \tag{22}$$

where $\hat{X}_n = X_n - \hat{\mu}_0$. The detector is built from the (squared) residuals $\hat{\epsilon}_n$, as:

$$\frac{1}{\sqrt{m}} TS_L^{on}(m, l) = \frac{1}{\sqrt{m}\hat{\eta}_m} \left| \sum_{n=m+1}^{m+l} \hat{\epsilon}_n^2 - \sum_{n=1}^{m} \hat{\epsilon}_n^2 \right|, \tag{23}$$

where $\hat{\eta}_m^2$ is a weakly consistent estimator of the moment $\eta_m^2 = E\left[\left(\epsilon_m^2 - \sigma_m^2\right)^2\right]$.

Finally, the boundary function is expressed as $b_L = cv_L^{on} g_L$, where $g_L = \left(1 + \frac{l}{m}\right)\left(\frac{l}{m+l}\right)^{\gamma}$, $\gamma \in [0, 1/2)$ and the critical value is obtained according to [49] as

$$\begin{aligned} \lim_{m\to\infty} Pr\{\tau(m) < \infty\} &= \lim_{m\to\infty} Pr\left\{\frac{TS_L^{on}}{g_L} \geq c_L^{on}(\alpha)\right\} \\ &= Pr\left(\sup_{n\in(0,1)} \frac{|W(n)|}{n^{\gamma}} \geq cv_L^{on}(\alpha)\right) = \alpha. \end{aligned} \tag{24}$$

C. NONLINEAR (NL) PARAMETRIC APPROACH USING A GENERALIZED AUTOREGRESSIVE CONDITIONAL HETEROSKEDASTICITY (GARCH) MODEL

A time series $\{X_n : n \in \mathbb{Z}\}$ follows the GARCH(p, q) process, if,

$$\begin{aligned} X_n &= \sigma_n \epsilon_n, \\ \sigma_n^2 &= \omega_n + \sum_{i=1}^{q} \alpha_{in} X_{n-i}^2 + \sum_{j=1}^{p} \beta_{jn} \sigma_{n-j}^2, \end{aligned}$$

where $\omega_n > 0$, $\alpha_{in}, \beta_{jn} \geqslant 0$ and $\{\epsilon_n : n \in \mathbb{Z}\}$ is a sequence of i.i.d r.v. with $E(\epsilon_1) = 0$ and $E(\epsilon_1^2) = 1$. We estimate the set of parameters θ_m during the initial training period, denoted in the following by $\theta_0 = (\omega_0, \alpha_{10}, , \cdots, \alpha_{q0}, \beta_{10}, \cdots, \beta_{p0})$; the estimation is performed by applying the Gaussian maximum-likelihood estimator (GMLE) $\hat{\theta}_0$ of θ_0 on the last m observations, as proposed in [53]. The GMLE function is given by

$$F_m(\theta; X_1, \cdots, X_m) = \prod_{n=1}^{m} \frac{1}{\sqrt{2\pi\hat{\sigma}_n^2}} \exp\left(-\frac{X_n^2}{2\hat{\sigma}_n^2}\right), \tag{25}$$

where $\hat{\sigma}_n^2$ are constructed recursively, as,

$$\hat{\sigma}_n^2 = \omega_n + \sum_{i=1}^{q} \alpha_{in} X_{n-i}^2 + \sum_{j=1}^{p} \beta_{jn} X_{n-j}^2. \tag{26}$$

Then, the GMLE of θ_m is,

$$\begin{aligned} \hat{\theta}_m &= \underset{\theta\in\Theta}{\operatorname{argmax}}\, F_m(\theta; X_1, \cdots, X_m) \\ &= \underset{\theta\in\Theta}{arg\,min}\, \frac{1}{m} \sum_{n=1}^{m} \left(\frac{X_n^2}{\hat{\sigma}_n^2} + \ln(\hat{\sigma}_n^2)\right). \end{aligned} \tag{27}$$

The residuals of the GARCH process are subsequently obtained from the GMLE as

$$\hat{\epsilon}_n = \frac{X_n}{\hat{\sigma}_n(\hat{\theta}_m)}. \tag{28}$$

Based on the (squared) residuals, the test statistic is described as in [54],

$$TS_{NL}^{on}(m, l) = \sqrt{\frac{m}{\operatorname{Var}(\hat{\epsilon}_m^2)}} \left| \frac{1}{l} \sum_{n=1}^{l} \hat{\epsilon}_n^2 - \frac{1}{m} \sum_{n=1}^{m} \hat{\epsilon}_n^2 \right|, \tag{29}$$

where $\operatorname{Var}(\hat{\epsilon}_n^2)$ denotes the variance of the squared residuals of the training period, i.e., $\operatorname{Var}(\hat{\epsilon}_m^2) = E(\hat{\epsilon}_m^4) - \left(E\left(\hat{\epsilon}_m^2\right)\right)^2$.

Considering the boundary function $b_{NL} = cv_{NL}^{on} g_{NL}$, we choose to work with $g_{NL} = 1$ as in [50]; consequently, the critical value is given by

$$\lim_{m\to\infty} Pr\{\tau(m) < \infty\} = \lim_{m\to\infty} Pr\left\{TS_{NL}^{on} \geq cv_{NL}^{on}(\alpha)\right\} = Pr\left(\sup_{n\in(0,1)} |W(n)| \geq cv_{NL}^{on}(\alpha)\right) = \alpha. \quad (30)$$

D. EVALUATION OF THE CRITICAL VALUES FOR THE CPS TESTS

The on-line critical values for the three procedures are estimated using Monte Carlo simulations, similarly to the off-line case, considering that

$$cv_{NP}^{on}(\alpha) = \sup_{n\in[0,1]} \left(\frac{H}{1+H}\right)^{\frac{1}{2}-\gamma} \frac{|W(n)|}{n^{\gamma}}, \quad (31)$$

$$cv_{L}^{on}(\alpha) = \sup_{n\in(0,1)} \frac{|W(n)|}{n^{\gamma}}, \quad (32)$$

$$cv_{NL}^{on}(\alpha) = \sup_{n\in(0,1)} |W(n)|. \quad (33)$$

With respect to the estimation of the magnitude of a detected CP denoted by cp_{on}^{*}, in the NP scenario, we estimate the deviation of the variance in pre-CP and post-CP data by comparing the variance of a pre-determined historical subsample, $\mathrm{Var}(X_{m_s} : X_{cp_{on}^{*}-h})$ to the variance "in the range" of the detected CP as $\mathrm{Var}(X_{cp_{on}^{*}-h} : X_{cp_{on}^{*}+h})$, accounting for the fact that a time lag $\pm h$ is required to establish the presence of an actual change.

We finally propose an alternative scheme to predict the post CP behavior in the case of a parametric model. We apply the parametric model (ARMA or GARCH) on the time horizon $t_{cp_{on}^{*}-h}, \ldots, t_{cp_{on}^{*}}$, in which we assume that the actual change has already occurred. Thus, a well defined subsample is provided to fit the model parameters and predict the next values using this adaptive model.

V. PERFORMANCE EVALUATION OF THE VARIANCE CP DETECTION APPROACHES ON SYNTHETIC DATA

In this Section, we evaluate the performance of the integrated algorithm with the three aforementioned variations of the on-line phase – NP, L and NL, – on two sets of synthetic data. In further detail, we report the results of Monte Carlo simulations using either an ARMA(1,1) or a GARCH(1,1) process to generate the time series; as a reminder, both ARMA and GARCH are well known models that have been shown to fit well video content popularity dynamics (see section II).

The synthetic sample size under consideration is $N = 1000$ while we introduce a variance CP at $cp^{*} = 500$; this is achieved by transforming the initial parameters vector of the chosen model. Evaluations are conducted based on 1000 repetitions for a significance level $\alpha = 0.01$. In all tests we set the beginning of the monitoring period at $m_s = 200$, the monitoring window length at $l = 100$ and the minimum interval between two successive CPs at $d = 80$ (this latter choice is justified by experiments with real data that will be presented in Section V). We experiment with two values for the sensitivity parameter $\gamma \in \{0, 0.25\}$ (as a reminder, γ only affects cv_{NP}^{on} and cv_{L}^{on}, see (31) and (32)).

We first evaluate the performance of the three alternative on-line procedures in the integrated algorithm, for a wide range of ARMA(1,1) models. We recall that the variance of an ARMA(1,1) model depends on the model parameters ϕ_i, θ_i and the variance of the error terms σ_i^2, i.e.,

$$\mathrm{Var}(X_n) = \frac{\left(1 + 2\phi_i\theta_i + \theta_i^2\right)\sigma_i^2}{1 - \phi_i^2}.$$

We consider a change by transforming the time series model defined by the parameter vector β_0 to one of the vectors $\beta_i, i = 1, 2, 3, 4$.

- Model 0: $\beta_0 = (\phi_0, \theta_0, \sigma_0) = (0.4, 0.2, 0.5)$,
- Model 1: $\beta_1 = (0.4, 0.2, 1)$,
- Model 2: $\beta_2 = (0.3, 0.3, 1.5)$,
- Model 3: $\beta_3 = (0.5, 0.3, 1.5)$,
- Model 4: $\beta_4 = (0.4, 0.2, 2)$.

We use Model 0 as the baseline. In Model 1 a small change in the error variance is introduced, which increases the uncertainty. Models 2 and 3 lead to medium changes in the variance and also transform the dependence structure between the r.v. On the other hand in Model 4 a large change is introduced by increasing the uncertainty.

In Table 1 we report the results of the simulation study. We depict the aggregate percentage of the CPs over the multitude of the simulations. For every test and each iteration we calculate the exact number of CPs detected:

- 0 when no CPs are detected, denoting the percentage of false negatives in all cases but the first (in which β_0 does not change); in this latter case it corresponds to the true success rate;
- 1 when a single CP is detected, denoting the true success rate in all cases but the first, in which it corresponds to a false positive rate;
- > 1 when more than one CPs are detected, denoting the percentage of false positives, in all cases other than the first. To obtain the overall false positive percentage, this value needs to be added to the false positive percentage above.

Furthermore, we denote by $\hat{cp}^{*}$ the median of the time instance of the identification of the true CP, evaluated in all cases but the first. The closest this number to the true point of the CP at 500, the quicker the detection and the better the responsiveness of the integrated algorithm.

Initially, we discuss the impact of the choice of the sensitivity parameter γ in the L and NP approaches. Studying Table 1, we conclude that $\gamma = 0$ is the most reasonable choice in the case of medium or more significant changes in the variance, since it leads to significantly lower false positive rates. On the other hand, in the case of only small changes in the variance, captured in our study in the

TABLE 1. Results from an ARMA generating process and for one change in the variance.

		ARMA(1,1)											
β	γ	Non parametric approach (NP)				ARMA approach (L)				GARCH approach (NL)			
		Detected CPs			$\hat{cp}^*$	Detected CPs			$\hat{cp}^*$	Detected CPs			$\hat{cp}^*$
		0	1	>1	med	0	1	>1	med	0	1	>1	med
β_0	0	**0.99**	0.01	0	-	**0.99**	0.01	0	-	**0.98**	0.02	0	-
	0.25	**0.95**	0.05	0	-	**0.98**	0.02	0	-				
β_1	0	0.49	**0.5**	0.01	-	0.48	**0.52**	0	554	0.74	**0.26**	0	-
	0.25	0.18	**0.76**	0.06	549	0.07	**0.93**	0	548				
β_2	0	0.04	**0.94**	0.02	550	0.03	**0.95**	0.02	546	0.15	**0.83**	0.02	549
	0.25	0	**0.93**	0.07	531	0	**0.96**	0.04	521				
β_3	0	0.01	**0.96**	0.03	536	0.01	**0.98**	0.01	535	0	**0.97**	0.03	548
	0.25	0	**0.92**	0.08	521	0	**0.97**	0.03	521				
β_4	0	0	**0.97**	0.03	533	0	**0.99**	0.01	530	0.01	**0.97**	0.02	544
	0.25	0	**0.93**	0.07	519	0	**0.97**	0.03	513				

TABLE 2. Results from a GARCH generating process and for one change in the variance.

		GARCH(1,1)											
θ	γ	non parametric approach (NP)				ARMA approach (L)				GARCH approach (NL)			
		Detected CPs			$\hat{cp}^*$	Detected CPs			$\hat{cp}^*$	Detected CPs			$\hat{cp}^*$
		0	1	>1	med	0	1	>1	med	0	1	>1	med
θ_0	0	**0.85**	0.15	0	-	**0.75**	0.25	0	-	**0.9**	0.1	0	-
	0.25	**0.65**	0.35	0	-	**0.42**	0.58	0	-				
θ_1	0	0.16	**0.8**	0.04	527	0.03	**0.77**	0.23	528	0.04	**0.92**	0.04	550
	0.25	0	**0.87**	0.13	521	0	**0.6**	0.4	515				
θ_2	0	0.03	**0.87**	0.1	524	0.01	**0.76**	0.23	521	0.01	**0.93**	0.06	544
	0.25	0.01	**0.85**	0.14	516	0	**0.56**	0.44	510				
θ_3	0	0	**0.93**	0.07	511	0	**0.7**	0.3	511	0	**0.93**	0.07	531
	0.25	0	**0.81**	0.19	508	0	**0.58**	0.42	505				

transformation from the β_0 to the β_1 model, a higher value of γ is needed (intuitively, for smaller changes a larger sensitivity is required). Therefore, depending on whether smaller or larger deviations need to be rapidly detected we can fine-tune the value of γ. For the sake of simplicity, in the following we focus on $\gamma = 0$ (larger deviations).

According to Table 1, the three approaches provide appropriate empirical sizes, and the false alarm rates are in all cases close to the significance level $\alpha = 0.01$. Overall the *L* procedure outperforms the *NP* and the *NL*, both in terms of the true alarm rates as well as in terms of the detection time; this is intuitive as in this first experiment the underlying process is generated by a linear ARMA(1,1) model and therefore a linear parametric model is excellently suited to capture the underlying dynamics. Furthermore, comparing the *NP* and the *NL* approaches, Table 1 illustrates that the *NP* is more sensitive than the *NL* approach, leading to more accurate detection for small changes at the cost of increased false positive rates in the case of larger changes. The opposite is true for the *NL* approach that appears to be more "conservative". Moreover, the fact that the *NP* procedure is statistically more sensitive leads to a quicker detection of a CP as captured through $\hat{cp}^*$.

We proceed to the more challenging case of a GARCH(1,1) generating model, with parameter vector $\theta_i = (\omega_i, \alpha_i, \beta_i)$ that fully describes the model and unconditional variance,

$$\mathrm{Var}(X_n) = \frac{\omega_i}{(1-\alpha_i-\beta_i)}.$$

To examine the alarm rates we assume the following models,

- Model 0: $\theta_0 = (\omega_0, \alpha_0, \beta_0) = (0.05, 0.4, 0.3)$,
- Model 1: $\theta_1 = (0.5, 0.2, 0.1)$,
- Model 2: $\theta_2 = (0.5, 0.3, 0.2)$,
- Model 3: $\theta_3 = (1, 0.3, 0.2)$.

GARCH is a varying volatility model, allowing volatility changes over time. Being more elaborate and complex in terms of the dependence of the variance on the model parameters, the higher false alarm and the lower true alarm rates in Table 2 are reasonable. In this case, the *L* procedure seems fully inappropriate irrespective of the choice of $\gamma = 0$ or $\gamma = 0.25$, suffering from very high false positive rates, since constant variance is assumed. The *NL* procedure, as expected, surpasses both the *L* and the *NP* procedures, as it is excellently suited to capture the GARCH process. More specifically, the true alarm rate estimation is stable for the different magnitudes of changes, with a detection time lag

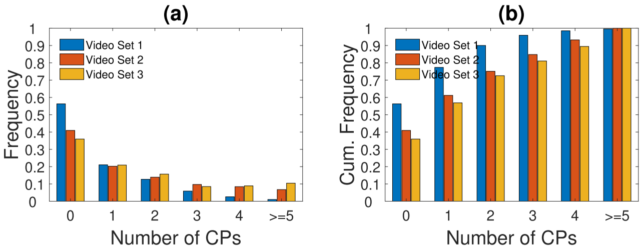

FIGURE 3. Estimated a) frequency and b) cumulative frequency of the number of CPs per time series, for three different Video Sets.

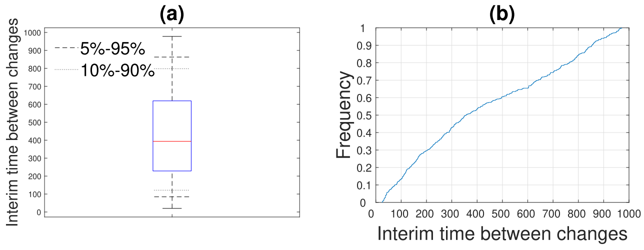

FIGURE 4. Interim time between consecutive CPs: a) Boxplot including the interval (5% – 95%) (dashed line) and (10% – 90%) interval (dotted line), b) Cumulative frequency for the interim time of consecutive CPs.

ranging from 50 instances for small changes to 31 instances for larger changes. On the other hand, the *NP* procedure appears to capture well the actual changes for $\gamma = 0$, with success rates relatively close to the those of the *NL* procedure, especially for medium/large changes. However, for $\gamma = 0.25$, the approach leads to ineligible false positive rates, despite the fact that it can identify small changes more efficiently. The *NP* method also achieves faster detection of changes, with $\hat{cp}^*$ ranging from 5 to 28 time instances.

Based on the analysis of the Monte Carlo results for the three procedures under the two different time series generating models, we can synthesize our overall conclusions in the following two points:

1) The *NL* and the *NP* approaches adapt better to a wider range of models and underlying assumptions; if there are indications of a highly nonlinear underlying procedure the *NP* approach could render better results;
2) The *L* approach is strongly related to the ARMA model assumptions and therefore it is advisable to be applied only if these can be readily shown to hold.

VI. ILLUSTRATION OF THE INTEGRATED ALGORITHM USING REAL DATA

Finally, we study the performance of the proposed algorithms on monitoring real YouTube video traces provided within the framework of the CONGAS project [55]; the dataset consists of 882 videos traces and the observation period is of $N = 1000$ time instances.

In this Section, we only adopt the non parametric (*NP*) and the GARCH (*NL*) approaches. We exclude the ARMA (*L*) approach from the evaluation, based on the conclusions of the previous Section. We work with the centered simple returns of the content popularity time series,

$$Y_n = (X_{n+1} - X_n) - \frac{1}{900}\sum_{n=1}^{900}(X_{n+1} - X_n), \quad n = 1, \ldots, 900$$

and then apply the methods on Y_n.

In order to clarify some general characteristics of the dataset, in terms of changing content dynamics, we first apply the off-line algorithm to the video traces. In Fig. 3, we consider three video sets; Video Set 1 contains the whole dataset, Video Set 2 contains the videos with average number of visits $\mathrm{E}((Y(1) : Y(1000)) \geq 10$ and Video Set 3 contains the videos with average number of visits greater or equal to 20. Fig. 3, depicts a high percentage of rejecting the H_0 hypothesis, for a significance level of $\alpha = 0.05$. Especially for the Video Sets 1 and 2, the rejection of the assumption of normal behavior exceeds 60% and 65% of the time series, respectively. This result confirms that a sufficiently high number of time series provide content popularity anomalies, for example in Video Set 3, in 10% of the cases there are over than four CPs per time series. This small analysis confirms the suitability of change point analysis as a viable approach for the detection of changes in video content popularity dynamics.

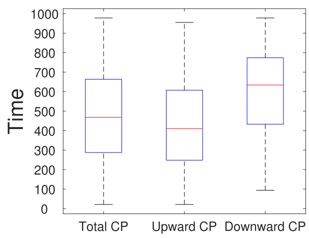

FIGURE 5. Boxplot of the number of upward and downward CPs, per time series.

Subsequently, in Fig. 4, we analyze the interim time between consecutive CPs. The respective boxplot diagrams illustrate the existence of sufficiently large intervals between consecutive changes; this fact supports our subtle assumption in Section III regarding the existence of a sufficient gap between two consecutive CPs (e.g., > 80 instances). In particular, 90% and 95% of the intervals correspond to consecutive CPs exceeding 100 and 80 time instances, respectively. This outcome assures that a sufficiently large training window after a detected change can be applied, denoted by the parameter d.

Additionally, Fig. 5, illustrates the time instances of upward (increase in volatility) and downward changes (decrease in volatility) in the form of a boxplot. It is shown that upward changes occur earlier in time than downward changes.

We consider now the performance of the on-line approach, by illustrating the estimated CPs in the second order characteristics of different time series. We choose the beginning of the monitoring period at $m_s = 200$, the sensitivity parameter

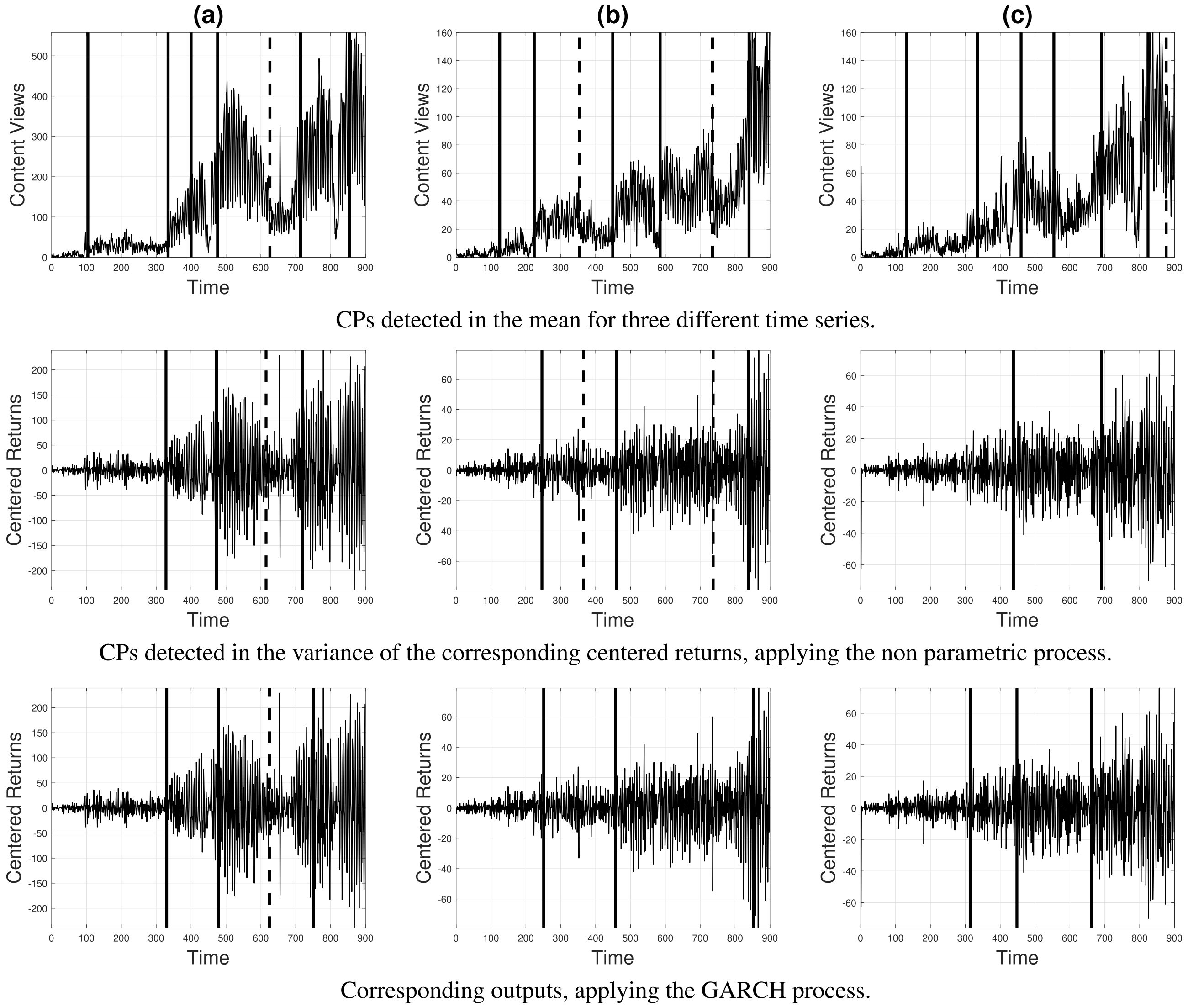

FIGURE 6. **CPs detected in the mean (first row) and variance (second and third row) for three different content views time series. Solid and dashed lines represent an upward and a downward change, respectively.**

$\gamma = 0$ and the significance level $\alpha = 0.05$. To fit a GARCH(p, q) model we consider all the possible combinations of the $p, q = 1, \cdots, 4$ and choose the orders p, q that minimize the Akaike information criterion (AIC).

The corresponding results are depicted in Fig. 6 at the top of the next page. The first row of results represent the detected changes in the mean value by using the RCPD algorithm presented in [11]. In the second and third row the estimated CPs in the variance are depicted, for the same time series, by applying on the first order differences Y_n the non parametric (*NP*) approach and the GARCH (*NL*) approach, respectively. Solid lines represent upwards changes while dashed lines represent downward changes.

Firstly, we observe that the variance changes are closely connected to a corresponding mean change. In particular, variance changes are less in multitude and seem to be related to the most significant mean changes, which can be intuitively explained by considering that if the average number of views changes significantly, the variance in the number of views at the respective interval will follow a similar trend. The importance of jointly studying the changes in the mean and the variance value is also depicted in Fig. 6. For instance, in Fig. 6a, to describe or handle the content popularity dynamics it is crucial to estimate quickly the "explosion" in variance after time instances 500 or 700, that leads to a high instability of the values from the mean. On the other hand, variance "reduction" detection is also important, as it implies that values remain relatively constant, like in Fig. 6a between time instances 600 and 700.

Both the *NP* and the *NL* approaches provide similar results in terms of the number of CPs and the detection time of the estimated CPs. More precisely, in Fig. 6a, both procedures detect the same number of changes, while the *NP* method gives a slightly quicker detection.

Focusing on the capability of the proposed algorithm to estimate the magnitude of a detected CP, we use the GARCH model. We estimate the parameters of the model considering 10 time instances before the detected change and forecast the variance for 10 time instances after the CP. For the time series in Fig. 6b, the actual variance after each change is 7.92, 12.51 and 38.66, while the predicted variance values are 7.39, 13.52 and 39.24, respectively. As we observe, in this case the *NL* algorithm can efficiently describe the post change variance behavior.

In the future, we will develop a joint approach identifying CPs simultaneously in the first and the second order characteristics, providing an aggregated and compact view of content popularity dynamics.

VII. CONCLUSION AND FUTURE WORK

In this paper, we presented an integrated algorithm for the detection of changes in the variance of a time series. We proposed to combine an off-line approach during which algorithmic and model parameters are learned. Subsequently, during the on-line part of the algorithm, changes in the variance of the time series are identified using a stopping time procedure. Whenever the value of a test statistic surpasses a predefined critical value, a change is declared.

To develop the test statistic we proposed three different approaches: i) a non-parametric approach, ii) a parametric approach using an ARMA model, and, iii) a parametric approach using a nonlinear GARCH model. Our studies using synthetic data indicated that the ARMA parametric approach does not generalize well. Due to this fact, we only performed experiments on real data using the non-parametric and the GARCH approaches. We concluded that both can equally well identify large deviations in the variance and that in the general case the non-parametric approach can provide quicker detection of CPs in the datasets studied in this work. In the future, we will develop joint detectors for the mean and the variance of video content popularity.

REFERENCES

[1] P. Sermpezis and T. Spyropoulos, "Effects of content popularity on the performance of content-centric opportunistic networking: An analytical approach and applications," *IEEE/ACM Trans. Netw.*, vol. 24, no. 6, pp. 3354–3368, Dec. 2016.

[2] M. Z. Shafiq, A. R. Khakpour, and A. X. Liu, "Characterizing caching workload of a large commercial content delivery network," in *Proc. IEEE INFOCOM, 35th Annu. IEEE Int. Conf. Comput. Commun.*, San Francisco, CA, USA, Apr. 2016, pp. 1–9.

[3] M. A. Salahuddin, J. Sahoo, R. Glitho, H. Elbiaze, and W. Ajib, "A survey on content placement algorithms for cloud-based content delivery networks," *IEEE Access*, vol. 6, pp. 91–114, 2018.

[4] S. Li, J. Xu, M. Van Der Schaar, and W. Li, "Popularity-driven content caching," in *Proc. IEEE INFOCOM, 35th Annu. IEEE Int. Conf. Comput. Commun.*, San Francisco, CA, USA, Apr. 2016, pp. 1–9.

[5] Y. Liu, T. Zhi, H. Xi, X. Duan, and H. Zhang, "A novel content popularity prediction algorithm based on auto regressive model in information-centric IoT," *IEEE Access*, vol. 7, pp. 27555–27564, 2019.

[6] F. Guillemin, B. Kauffmann, S. Moteau, and A. Simonian, "Experimental analysis of caching efficiency for YouTube traffic in an ISP network," in *Proc. 25th Int. Teletraffic Congr. (ITC)*, Shanghai, China, 2013, pp. 1–9.

[7] W.-X. Liu, J. Zhang, Z.-W. Liang, L.-X. Peng, and J. Cai, "Content popularity prediction and caching for ICN: A deep learning approach with SDN," *IEEE Access*, vol. 6, pp. 5075–5089, 2018.

[8] Y. Zheng, L. Liu, L. Wang, and X. Xie, "Learning transportation mode from raw GPS data for geographic applications on the Web," in *Proc. 17th Int. Conf. World Wide Web (WWW)*, Beijing, China, Apr. 2008, pp. 247–256.

[9] J. Song, M. Sheng, T. Q. S. Quek, C. Xu, and X. Wang, "Learning-based content caching and sharing for wireless networks," *IEEE Trans. Commun.*, vol. 65, no. 10, pp. 4309–4324, Oct. 2019.

[10] F. Jiang, Z. Yuan, C. Sun, and J. Wang, "Deep Q-learning-based content caching with update strategy for fog radio access networks," *IEEE Access*, vol. 7, pp. 97505–97514, 2019.

[11] S. Skaperas, L. Mamatas, and A. Chorti, "Early video content popularity detection with change point analysis," in *Proc. IEEE Global Commun. Conf. (GLOBECOM)*, Abu Dhabi, UAE, Dec. 2018, pp. 1–7.

[12] S. Skaperas, L. Mamatas, and A. Chorti, "Real-time video content popularity detection based on mean change point analysis," *IEEE Access*, vol. 7, pp. 142246–142260, 2019.

[13] V. Jandhyala, S. Fotopoulos, I. Macneill, and P. Liu, "Inference for single and multiple change-points in time series," *J. Time Anal.*, vol. 34, pp. 423–446, May 2013.

[14] A. Aue and L. Horváth, "Structural breaks in time series," *J. Time Anal.*, vol. 34, no. 1, pp. 1–16, Jan. 2013.

[15] C. Callegari, A. Coluccia, A. D'Alconzo, W. Ellens, S. Giordano, M. Mandjes, M. Pagano, T. Pepe, F. Ricciato, and P. Zuraniewski, "A methodological overview on anomaly detection," in *Data Traffic Monitoring and Analysis*. Berlin, Germany: Springer-Verlag, 2013, pp. 148–183.

[16] V. Chandola, A. Banerjee, and V. Kumar, "Anomaly detection: A survey," *ACM Comput. Surv.*, vol. 41, no. 3, pp. 1–58, Jul. 2009.

[17] S. Aminikhanghahi and D. J. Cook, "A survey of methods for time series change point detection," *Knowl. Inf. Syst.*, vol. 51, no. 2, pp. 339–367, May 2017.

[18] D. Choudhary, A. Kejariwal, and F. Orsini, "On the runtime-efficacy trade-off of anomaly detection techniques for real-time streaming data," Oct. 2017, *arXiv:1710.04735*. [Online]. Available: https://arxiv.org/abs/1710.04735

[19] J. Chen and A. K. Gupta, *Parametric Statistical Change Point Analysis: With Applications to Genetics, Medicine, and Finance*. Cambridge, MA, USA: Birkhäuser 2000.

[20] G. Tsechpenakis, D. N. Metaxas, C. Neidle, and O. Hadjiliadis, "Robust online change-point detection in video sequences," in *Proc. Comput. Vis. Patt. Recog. Work.*, New York, NY, USA, Jun. 2006, p. 155.

[21] V. Konev and S. Vorobeychikov, "Quickest detection of parameter changes in stochastic regression: Nonparametric CUSUM," *IEEE Trans. Inf. Theory*, vol. 63, no. 9, pp. 5588–5602, Sep. 2017.

[22] A. G. Tartakovsky, A. S. Polunchenko, and G. Sokolov, "Efficient computer network anomaly detection by change point detection methods," *IEEE J. Sel. Topics Signal Process.*, vol. 7, no. 1, pp. 4–11, Feb. 2013.

[23] A. Soule, K. Salamatian, and N. Taft, "Combining filtering and statistical methods for anomaly detection," in *Proc. 5th ACM SIGCOMM Conf. Internet Meas. (IMC)*, New York, NY, USA, Oct. 2005, p. 31.

[24] I. Nevat, D. M. Divakaran, S. G. Nagarajan, P. Zhang, L. Su, L. L. Ko, and V. L. L. Thing, "Anomaly detection and attribution in networks with temporally correlated traffic," *IEEE/ACM Trans. Netw.*, vol. 26, no. 1, pp. 131–144, Feb. 2018.

[25] G. Thatte, U. Mitra, and J. Heidemann, "Parametric methods for anomaly detection in aggregate traffic," *IEEE/ACM Trans. Netw.*, vol. 19, no. 2, pp. 512–525, Apr. 2011.

[26] S. Lee, J. Ha, O. Na, and S. Na, "The cusum test for parameter change in time series models," *Scandin. J. Statist.*, vol. 30, no. 4, pp. 781–796, Dec. 2003.

[27] I. Berkes, E. Gombay, L. Horváth, and P. Kokoszka, "Sequential change-point detection in GARCH (p, q) models," *Econ. Theory*, vol. 20, no. 6, pp. 1140–1167, Dec. 2004.

[28] Y. Borghol, S. Mitra, S. Ardon, N. Carlsson, D. Eager, and A. Mahanti, "Characterizing and modelling popularity of user-generated videos," *Perform. Eval.*, vol. 68, no. 11, pp. 1037–1055, Nov. 2011.

[29] G. Szabo and B. A. Huberman, "Predicting the popularity of online content," *Commun. ACM*, vol. 53, no. 8, pp. 80–88, Aug. 2010.

[30] J. Xu, M. Van Der Schaar, J. Liu, and H. Li, "Forecasting popularity of videos using social media," *IEEE J. Sel. Topics Signal Process.*, vol. 9, no. 2, pp. 330–343, Mar. 2015.

[31] P. Gill, M. Arlitt, Z. Li, and A. Mahanti, "Youtube traffic characterization: A view from the edge," in *Proc. 7th ACM SIGCOMM Conf. Internet Meas.*, San Diego, CA, USA, 2007, pp. 15–28.

[32] S. Tanwir and H. Perros, "A survey of VBR video traffic models," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 4, pp. 1778–1802, 4th Quart., 2013.

[33] M. Tsagkias, W. Weerkamp, and M. De Rijke, "News comments: Exploring, modeling, and online prediction," in *Proc. Eur. Conf. Inform. Retr.*, Milton Keynes, U.K., Mar. 2010, pp. 191–203.

[34] M. Cha, H. Kwak, P. Rodriguez, Y.-Y. Ahn, and S. Moon, "Analyzing the video popularity characteristics of large-scale user generated content systems," *IEEE/ACM Trans. Netw.*, vol. 17, no. 5, pp. 1357–1370, Oct. 2009.

[35] T. Trzcinski and P. Rokita, "Predicting popularity of online videos using support vector regression," *IEEE Trans. Multimedia*, vol. 19, no. 11, pp. 2561–2570, Nov. 2017.

[36] S. Romano and H. Elaarag, "A neural network proxy cache replacement strategy and its implementation in the Squid proxy server," *Neural Comput. Appl.*, vol. 20, no. 1, pp. 59–78, Feb. 2011.

[37] W. Ali and S. M. Shamsuddin, "Intelligent client-side Web caching scheme based on least recently used algorithm and neuro-fuzzy system," in *Proc. Int. Symp. Neural Netw. (ISNN)*, Wuhan, China, May 2009, pp. 70–79.

[38] G. Gursun, M. Crovella, and I. Matta, "Describing and forecasting video access patterns," in *Proc. IEEE INFOCOM*, Shanghai, China, Apr. 2011, pp. 16–20.

[39] D. Niu, Z. Liu, B. Li, and S. Zhao, "Demand forecast and performance prediction in peer-assisted on-demand streaming systems," in *Proc. IEEE INFOCOM*, Shanghai, China, Apr. 2011, pp. 421–425.

[40] C. Katris and S. Daskalaki, "Generation of synthetic video traffic using time series," *Simul. Model. Pract. Theory*, vol. 75, pp. 127–145, Jun. 2017.

[41] B. Zhou, D. He, Z. Sun, and W. H. Ng, "Network traffic modeling and prediction with ARIMA/GARCH," in *Proc. HET-NETs Conf.*, IIkley, U.K., Jul. 2005, pp. 1–10.

[42] C. Katris and S. Daskalaki, "Dynamic bandwidth allocation for video traffic using FARIMA-based forecasting models," *J Netw. Syst. Manage.*, vol. 27, no. 1, pp. 39–65, Jan. 2019.

[43] D. Niu, H. Xu, and B. Li, "Resource auto-scaling and sparse content replication for video storage systems," *ACM Trans. Model. Perform. Eval. Comput. Syst.*, vol. 2, no. 4, pp. 1–30, Nov. 2017.

[44] P. Valsamas, S. Skaperas, and L. Mamatas, "Elastic content distribution based on unikernels and change-point analysis," in *Proc. 24th Eur. Wireless Conf. (EW)*, Catania, Italy, 2018, pp. 1–7.

[45] A. Aue, S. Hörmann, L. Horváth, and M. Reimherr, "Break detection in the covariance structure of multivariate time series models," *Ann. Statist.*, vol. 37, no. 6B, pp. 4046–4087, Dec. 2009.

[46] D. W. K. Andrews, "Heteroskedasticity and autocorrelation consistent covariance matrix estimation," *Econometrica*, vol. 59, no. 3, pp. 817–858, May 1991.

[47] C. Inclan and G. C. Tiao, "Use of cumulative sums of squares for retrospective detection of changes of variance," *J. Amer. Stat. Assoc.*, vol. 89, no. 427, pp. 913–923, Sep. 1994.

[48] K. Pape, D. Wied, and P. Galeano, "Monitoring multivariate variance changes," *J. Empirical Finance*, vol. 39, pp. 54–68, Dec. 2016.

[49] A. Aue, C. Dienes, S. Fremdt, and J. Steinebach, "Reaction times of monitoring schemes for ARMA time series," *Bernoulli*, vol. 21, no. 2, pp. 1238–1259, May 2015.

[50] O. Na, Y. Lee, and S. Lee, "Monitoring parameter change in time series models," *Stat. Methods Appl.*, vol. 20, no. 2, pp. 171–199, Jun. 2011.

[51] J. Davidson, *Stochastic Limit Theory: An Introduction for Econometricians*. New York, NY, USA: Oxford Univ. Press, 1994.

[52] D. Wied, M. Arnold, N. Bissantz, and D. Ziggel, "A new fluctuation test for constant variances with applications to finance," *Metrika*, vol. 75, no. 8, pp. 1111–1127, Nov. 2012.

[53] C. Francq and J.-M. Zakoïan, "Maximum likelihood estimation of pure GARCH and ARMA-GARCH processes," *Bernoulli*, vol. 10, no. 4, pp. 605–637, Aug. 2004.

[54] W. L. N. Leung, S. Him, and C. Y. Yau, "Sequential change-point detection in time series models based on pairwise likelihood," *Statistica Sinica*, vol. 27, no. 2, pp. 575–605, Apr. 2017.

[55] F. De Pellegrini, M. Zeni, and D. Miorandi, "YOUStatAnalyzer: A tool for analysing the dynamics of YouTube content popularity," in *Proc. 7th Int. Conf. Perform. Eval. Methodol. Tools*, Torino, Italy, Dec. 2013, pp. 286–289.

SOTIRIS SKAPERAS (Student Member, IEEE) received the B.Sc. degree in mathematics and the M.Sc. degree in statistics and modeling from the Department of Mathematics, Aristotle University of Thessaloniki, Greece, in 2013 and 2016, respectively. He is currently pursuing the Ph.D. degree in the area of resource management in 5G networks with the Department of Applied Informatics, University of Macedonia, Thessaloniki. He involved in the areas of resource allocation/load balancing for 5G networks using time-series/change point analysis and stochastic modeling.

LEFTERIS MAMATAS (Member, IEEE) received the Diploma and Ph.D. degrees from the Department of Electrical and Computer Engineering, Democritus University of Thrace, Greece, in 2003 and 2008, respectively. He was a Researcher with the University College London, U.K., the Space Internetworking Center/Democritus University of Thrace, Greece, and the DoCoMo Euro-Labs, Munich, Germany. He is currently an Assistant Professor with the Department of Applied Informatics, University of Macedonia, Thessaloniki, Greece, where he leads the Softwarized and Wireless Networks Research Group. He has published more than 60 articles in international journals and conferences. His research interests include the areas of software-defined networks, the Internet of Things, 5G networks, and multiaccess edge computing. He participated in many international research projects, such as NECOS (H2020), FED4FIRE+OC4 (H2020), WiSHFUL OC2 (H2020), MONROE OC2 (H2020), Dolfin(FP7), UniverSELF (FP7), and Extending Internet into Space (ESA). He served as the General Chair for the WWIC2016 Conference and the INFO-COM SWFAN 2016 Workshop and as the TPC Chair for the INFOCOM SWFAN 2017, E-DTN 2009, and IFIP WWIC 2012 conferences/workshops. He is also a Guest Editor for the Journal *Ad Hoc Networks* (Elsevier).

ARSENIA CHORTI (Member, IEEE) received the M.Eng. degree in electrical and electronic engineering from the University of Patras, Greece, the D.E.A. degree in electronics from the University Pierre et Marie Curie, Paris VI, France, and the Ph.D. degree in electrical engineering from Imperial College London, U.K., in November 2005. She held postdoctoral positions with the University of Southampton, U.K., Technical University of Crete, Greece, and University College London, U.K., from 2005 to 2008. She served as a Senior Lecturer in communications for Middlesex University, U.K., from December 2008 to April 2010. From 2010 to 2013, she was a Marie Curie IOF Researcher with Princeton University, NJ, USA, and with the Institute of Computer Science-FORTH, Greece. From 2013 to 2017, she was a Lecturer with the University of Essex, U.K. She is currently an Associate Professor with ETIS, UMR 8051, University Paris Seine, University Cergy-Pontoise, ENSEA, CNRS, Cergy, France. She is also a Visiting Research Fellow with the University of Essex, U.K. Her work has so far been disseminated in more than 60 journals and international conferences and one book.