IEEE Access
practical innovations : open solutions

Received October 24, 2013, accepted December 17, 2013, date of current version January 14, 2014.

Digital Object Identifier 10.1109/ACCESS.2013.2297736

Parsimonious Network Traffic Modeling By Transformed ARMA Models

MARKUS LANER (Student Member, IEEE), PHILIPP SVOBODA (Member, IEEE), AND MARKUS RUPP (Senior Member, IEEE)

Institute of Telecommunications, Vienna University of Technology, Vienna 1040, Austria

Corresponding author: M. Laner (mlaner@nt.tuwien.ac.at)

ABSTRACT Generating synthetic data traffic, which statistically resembles its recorded counterpart is one of the main goals of network traffic modeling. Equivalently, one or several random processes shall be created, exhibiting multiple prescribed statistical measures. In this paper, we present a framework enabling the joint representation of distributions, autocorrelations and cross-correlations of multiple processes. This is achieved by so called transformed Gaussian autoregressive moving-average models. They constitute an analytically tractable framework, which allows for the separation of the fitting problems into subproblems for individual measures. Accordingly, known fitting techniques and algorithms can be deployed for the respective solution. The proposed framework exhibits promising properties: 1) relevant statistical properties such as heavy tails and long-range dependences are manageable; 2) the resulting models are parsimonious; 3) the fitting procedure is fully automatic; and 4) the complexity of generating synthetic traffic is very low. We evaluate the framework with traced traffic, i.e., aggregated traffic, online gaming, and video streaming. The queueing responses of synthetic and recorded traffic exhibit identical statistics. This paper provides guidance for high-quality modeling of network traffic. It proposes a unifying framework, validates several fitting algorithms, and suggests combinations of algorithms suited best for specific traffic types.

INDEX TERMS Traffic modeling, transformed Gaussian, ARMA model, parsimoniousness.

I. INTRODUCTION

Emulation of data traffic facilitates the verification and testing of network equipment. For this purpose real traffic is recorded, respective properties are abstracted and modeled. The represented characteristic (physical quantity) varies with the field of application: (i) Classic queuing theory deals with arrivals of packets [1], [2]. (ii) In multimedia streaming the traffic is commonly characterized by the size of the video frames [3], [4]. (iii) For online-gaming traffic both, the Internet Protocol (IP) Packet Size (PS) as well as the Inter Packet-Arrival Time (IAT), are jointly considered [5]. Stationary stochastic process(es) are the foundation for most modeling approaches within the field [6], [7]. A unified framework for describing data traffic is therefore feasible; a respective construction is the scope of this work.

The targeted generality complicates the definition of design goals; namely, it is not obvious how the quality of a unified framework can be assessed. Therefore, we list our priorities in the following. A traffic modeling framework should:

- enable to capture numerous statistics of the original processes, the more the better,
- be flexible enough to be applied for a variety of different traffic types (e.g., video, web, background),
- inherit simple (i.e., automated) fitting procedures,
- yield parsimonious models (in terms of model parameters) and
- facilitate the generation of synthetic traffic with low complexity.

In literature several properties of stochastic processes have been evidenced to cause relevant effects to traffic modeling [8]–[14]. In this work we focus on the most basic properties, namely:

- marginal Cumulative Distribution Function (CDF),
- Auto-correlation Function (ACF) and
- Cross-correlation Function (XCF).

The last point implies that multiple processes might be considered jointly, in order to achieve satisfactory modeling accuracy (e.g., PS and IAT).

The vast majority of publication considers one or several of the three mentioned statistics; few works investigate on higher order properties, such as bi-spectra [15] or joint distribution functions [16]. Such higher order properties are not within the scope of this work.

A. MOTIVATING EXAMPLE

We demonstrate the fact that the three basic statistics are utmost important for data traffic modeling on a simple example. A common problem statement in network performance testing is to assess the queueing response for given input traffic. Accordingly, we investigate the queueing response of synthetic data traffic, where CDF, ACF and XCF are incrementally introduced.

A queue with a single server and constant service time per byte is simulated (e.g., a communication link shows such a behavior). The input process consists of a packet stream, of which the statistical properties of the PS and the IAT are varied for different simulation runs by using the method presented within this work. The Complementary Cumulative Distribution Functions (CCDFs) of the queue length are depicted in Fig. 1. The different curves show the following scenarios: (i) constant IAT and gamma distributed PS, e.g., encountered for video traffic [17], [18], causing the shortest buffers – leftmost curve, (ii) additionally, the constant packet IAT is changed to an exponentially distributed IAT, (iii) auto-correlations are introduced to the PS process by an AR(1) filter, (iv) furthermore, the same auto-correlation structure is introduced to the IAT and (v) finally, strong negative cross-correlation between PS and IAT is imposed to the processes (i.e., big PSs coincide with small IATs), which is causing the longest buffers – rightmost curve. It is clearly visible in Fig. 1 that CDFs as well as ACFs and the XCF have an impact on the queue length, for which variations over more than two decades are observed. Conversely, for example, if the rightmost curve (v) would correspond to original measured traffic and one would model it by fitting only its CDFs (i.e., neglecting ACFs and XCF, corresponding to a renewal process), the queueing response of the model would correspond to the second curve from the left (ii); hence, the actual queueing response would be underestimated by about two magnitudes. Similar examples can further be constructed for modeling only the ACFs and only the XCF.

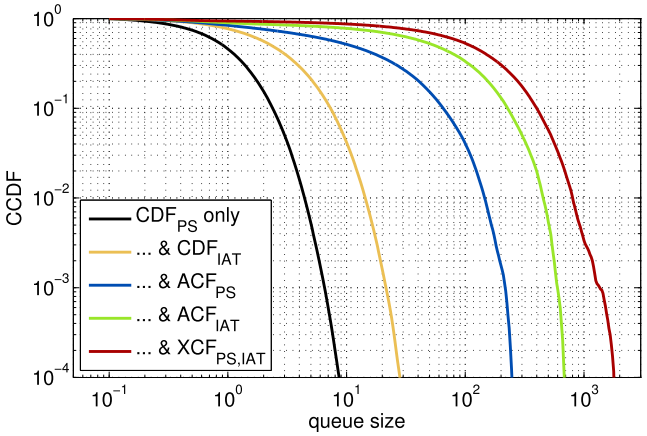

FIGURE 1. Queueing response on data traffic with different statistical properties of the respective packet size and packet inter-arrival time processes (utilization: 80%).

B. CONTRIBUTIONS

In this article we propose a novel modeling framework, a variant of so called Transformed Auto-Regressive Moving-Average (TARMA) processes, which is able to jointly characterize a broad range of CDFs, ACFs and XCFs for multiple random processes, as they typically appear in the context of network traffic modeling.

The proposed modeling framework consists of several known building blocks. The novelty is the specific selection and arrangement of these blocks. The framework is tailored to the characterization of network traffic and shows, that Auto-Regressive Moving-Average (ARMA) models, although rarely deployed in the community, are well suited for this task.

The key advantage of the framework is the separability of the model fitting problem into independent problems for CDFs, ACFs and XCFs. This feature enables parsimoniousness in the number of model parameters, since (i) parsimonious modeling procedures are known for each sub-problem and (ii) the number of parameters deployed for one sub-problem has no impact on the other sub-problems. To the best of our knowledge, this work is the first achieving this combination.

A summary and comparison of diverse fitting algorithms is given. Specific sequences of algorithms are suggested for individual traffic types. Fully automated fitting is feasible and demonstrated for (i) backbone traffic, (ii) online gaming and (iii) video streaming. Corresponding synthetic data traffic exhibits queueing responses very similar to its real counterpart.

C. ORGANIZATION OF THIS ARTICLE

In Sec. II we provide a general overview on traffic modeling categories and frameworks. Sec. III introduces the proposed modeling framework and explains how to generate synthetic network traffic from given model parameters. In Sec. IV we present and compare model fitting algorithms. We further suggest sets of algorithms for specific traffic types. Sec. V contains general remarks on the proposed framework. An evaluation with real data traffic is given in Sec. VI. Finally, Sec. VII concludes the article.

II. RELATED WORK

Traffic modeling is an active research topic since roughly three decades, [6], [7]. Most modeling approaches have in common that they model traffic streams as one or more stochastic processes. One simple approach is to assume one renewal process to be sufficient for representing all relevant properties of the traffic. In this case only the distribution of the process can be modeled. The respective methods for parameter estimation and synthetic traffic generation are well established in literature [19]–[21] and implemented in simulation tools. The distributions exhibit characteristics which impact on the network and queueing response of the

random processes; hence, respective modeling is justified. Such characteristics may be, for example, heavy-tails [22]–[24] or, more general, skewness [4], [17], [25], [18].

The assumption of independent identically distributed (i.i.d.) random processes (renewal processes) is, however, often violated for network traffic [8], [26]–[28]. This is especially the case if long-range dependencies and self-similarities occur [23], [29], [30]. Besides non-stationary modeling approaches (which are beyond the scope of this work), there are two standard classes of models for such temporal dependencies; namely, regression models (e.g., ARMA) and Markov models. ARMA models rely on linear filter theory and benefit from a long history with comprehensive literature [21] and [31]. However, they are only able to handle a limited class of distributions (e.g., normal distributions). Markov models, on the other hand, are a very general tool, able to model various discrete distributions and auto-correlations. For obtaining continuous distributions, as mostly required for traffic modeling, Markov models have to be extended to Markovian models (e.g., hidden Markov models, Markov modulated processes).

A further property of interest for traffic modeling is to capture cross-correlations in network traffic. This idea is natural when migrating from one to multiple random processes (i.e., physical quantities). Examples are (i) packet networking, where size (PS) and arrival time (IAT) have to be considered jointly, (ii) video streaming, where different video frame types (e.g., I,P and B in MPEG4) can be treated as individual random processes, and (iii) multiplayer online gaming, where one server issues multiple correlated packet streams to the individual players. Literature provides few examples for traffic models where cross-correlations were considered.

Summarizing, standard models are not able to represent broad ranges of CDFs, ACFs and XCFs jointly. This results in a variety of data traffic models, each of which designed for either a specific application or a specific network type. On the other hand, there are only few modeling approaches which are capable of jointly representing the above mentioned statistics. Those can be summarized in three categories: (i) Markovian models, (ii) TES models and (iii) transformed Gaussian ARMA models; they are summarized below and a respective comparison is given in Table 1, see [32]. Note, that all three modeling frameworks are highly sophisticated and have been extended by various authors. It is therefore difficult to provide a fair comparison which is generally valid, especially regarding the fitting accuracy and parsimoniousness. The modeling framework presented in this article belongs to the last category: Transformed Auto-Regressive Moving-Average (TARMA) models.

A. MARKOVIAN MODELS

This type of models is most often encountered in literature [33] and [34] within various different contexts, such as speech [35], video [17] and online gaming [36]. It comes in various flavors, for example, Markov Modulated Poisson Process (MMPP) or Markovian Arrival Process (MAP). They base on a hidden Markov chain generating state dependent arrivals, which are summarized to a common random process. This yields a highly flexible structure, which is able to characterize arbitrary CDFs and ACFs jointly. Furthermore, the resulting processes are fully analytically tractable. The drawback of this approach appears when the model is fitted to data; namely, the CDF and ACF have to be fitted jointly. This implies that the fitting process has to iterate between CDF and ACF, which is computationally intensive. Further, the number of parameters to be fitted for both CDF and ACF is coupled, which is not optimal from a parsimoniousness point of view. Recent work in the field tackles this problem and achieves good fitting performance with a low amount of model parameters [34]. Further, MAPs have been extended to capture the XCF of multiple processes in [33], where the authors accurately fit their model to various traffic types.

TABLE 1. Comparison of generic traffic models.

	Markovian	TES	TARMA
Statistical measure	CDF, ACF, XCF		
Heavy tailed distribution	approximations		
Long correlation	approximations		
Analytically tractable	✓	✓	✓
Queueing theoretic results	✓		
Separable fitting problems		✓	✓
Parsimonious	✓		✓
Automatic fitting	✓		✓
Fitting complexity	high	medium	low
Fitting acc. CDF smooth	high	high	high
Fitting acc. CDF peaky	high	n/a	medium
Fitting accuracy ACF	high	n/a	high
Fitting accuracy XCF	n/a	n/a	high
Sample generation complexity	$\mathcal{O}(N)$	$\mathcal{O}(N)$	$\mathcal{O}(N)$

B. TES MODELS

The acronym TES stands for *Transform Expand Sample*, an approach based on uniform random processes [37], [38]. The ACF and CDF are introduced to the process in two steps which are decoupled. This is due to the fact that auto-correlations can be introduced to a uniform random process without changing its distribution. The model benefits from the vast amount of available transformations from uniform distributions to any other type of distribution, which is the basis of all random numbers in computers [20]. Nevertheless, the method faces problems with the smoothness of the sample paths and with fitting auto-correlations at large lags, which requires interaction of the user during the fitting procedure.

C. TRANSFORMED GAUSSIAN ARMA MODELS

This category encompasses the framework presented below. It comprises various works from different fields of study [18], [39]–[42]. It bases on correlated Gaussian random processes which are warped by a memoryless non-linear transformation. The ACF and CDF are introduced in two decoupled steps, first the ACF, depolying regression models (e.g., ARMA models), then the CDF by a non-linearity. As both other approaches, transformed Gaussian models in its

general form are able to reproduce any desired CDF. Further, also a vast range of ACFs are captured, which can be fitted parsimoniously due to the evolved methods of linear system theory [19], [21], [31]. The ACF of the output process is further analytically tractable, which is in general computationally expensive. In order to overcome this problem, Hermitian polynomials are proposed as non-linearity [39], for which closed form solutions can be found for ACF and XCF after transformation (see [40, pp. 419–426], [43, pp. 132ff.], and [44, pp. 143ff.] and Annex I-A, Annex I-B). This approach is very flexible.

In the context of Internet traffic modeling, Auto-Regressive To Anything (ARTA) models have to be mentioned [41]. The authors show that the approach is suited for traffic modeling and provide a general analytical framework, by leaving the non-linearity unspecified. Recent work in the field of ARTA modeling extends this method to a combination of ARMA models with Markov models [45]. The authors show that ARMA processes are suitable for introducing correlations into phase-type distributions. Similar approaches appeared in video traffic modeling [4], [18], where the focus is mainly on the generation of correlated chi-squared and gamma processes. The method is referred to as Gaussian Auto-Regressive and Chi-Squared (GACS) models and is fully analytically tractable.

The generation of Gaussian random processes with cross-correlation is preferable over other methods because of the numerously available literature (see [21, p. 551ff.] and [31, p. 401ff.]) and the possibility of decoupling the fitting problem of ACFs and XCF. A theoretical framework for the generation of multiple random processes based on ARTA models is given in [46]. In the field of video modeling, the modeling of correlation coefficients (i.e., XCFs at lag 0) is treated in [47] and [48]; the results show an improvement over models which neglect cross-correlations. A recently presented method [16] additionally allows to fit multidimensional joint distribution functions of multiple random processes.

III. GENERATING TRAFFIC FROM TRANSFORMED ARMA MODELS

In this section, we explain the functional principle of the TARMA modeling approach. Further, the generation/emulation of data traffic from given model parameters is described, which may act as input for network simulations. For the rest of this section the physical quantities of the output processes $Z_i[n]$ are not specified; however, common examples are PS and IAT.

The proposed modeling approach allows for the joint representation of arbitrary CDFs, ACFs and XCFs. This is achieved by three sequential transformations of normal i.i.d. random processes, each of which being responsible for handling one of the above statistical measures. A corresponding block diagram for the generation of I output processes $Z_i[n]$ is depicted in Fig. 2. The four different types of blocks have the following functionalities:

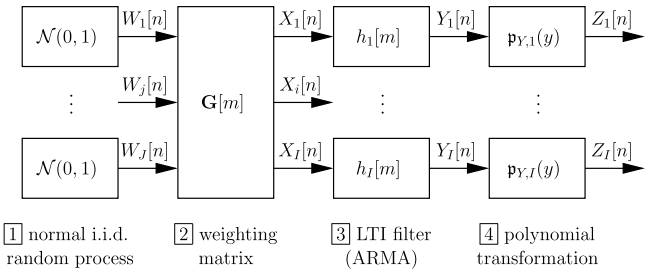

FIGURE 2. Block diagram of the proposed modeling approach for the generation of network source traffic, I inter-dependent random processes.

1. **Gaussian i.i.d. random process**. These blocks generate J independent normal random processes with zero mean and unit variance.
2. **Weighting matrix**. Matrix $\mathbf{G}[m]$ introduces cross-correlation to the I output processes $X_i[n]$.
3. **LTI filter (ARMA)**. The Linear Time Invariant (LTI) filters $h_i[m]$ introduce auto-correlations to the processes $Y_i[n]$.
4. **Polynomial transformation**. The memoryless polynomials $\mathfrak{p}_{Y,i}(y)$ shape the distributions of $Z_i[n]$.

Between the four blocks, the intermediate random processes $W_j[n]$, $X_i[n]$ and $Y_i[n]$ are observed; they can be interpreted as the passing of one random sample per time index n from one block to its successor. For ensuring that the mentioned blocks fulfill their task properly, the following restrictions have to be satisfied for the respective input processes:

1) The CDF of the process is normal (Gaussian) with zero mean and unit variance,
2) The ACF of the process is zero for all lags $m \neq 0$,
3) The XCF between the processes is zero.

For the process $W_j[n]$ all three conditions must apply, for $X_i[n]$ the first two conditions are necessary, leading to Eq. (13) and Eq. (14), and for $Y_i[n]$ only the first condition is required, leading to Eq. (6). Sticking to those requirements also guarantees that the fitting problems for CDF, ACF and XCF are separable, one of the key features of the proposed model.

In the following each functional block is described in detail. For convenience, a summary of the rest of this section is anticipated:

1) Samples of $Z_i[n]$ are generated by the consecutive accomplishment of Eq. (11), Eq. (5) and Eq. (1) on normal i.i.d. samples.
2) The Probability Density Function (PDF) of $Z_i[n]$ can be calculated, see Sec. III-A.
3) ACFs can be calculated by Eq. (8) and Eq. (3).
4) XCFs can be assessed by Eq. (12), Eq. (10) and Eq. (4).

This framework allows for the complete analytical tractability of the processes $Z_i[n]$. The description of the functional blocks of Fig. 2 is given in the following in reverse order. If used in a unique manner, index i of the random processes will be dropped.

A. MEMORYLESS POLYNOMIAL TRANSFORMATION

The last block of the chain, block 4, performs a polynomial transformation $\mathfrak{p}_Y(y)$ of the random process $Y[n]$, according to

$$Z[n] = \mathfrak{p}_Y(Y[n]) = \sum_{p=0}^{P} \alpha_p \cdot (Y[n])^p, \tag{1}$$

where P is the order of the polynomial $\mathfrak{p}_Y(y)$ and α_p are the coefficients of the polynomial for the power p. The goal of this transformation is to resemble a quantile transformation procedure. It enables the generation of a random variable $Z[n]$ with arbitrary distribution, from any other distribution of $Y[n]$ by mapping the corresponding percentiles to each other [19, p. 139]. This is according to

$$Z[n] = F_Z^{-1}(F_Y(Y[n])), \tag{2}$$

where $F_Y(\cdot)$ denotes the CDF of the random process $Y[n]$ and $F_Z^{-1}(\cdot)$ the inverse of the desired CDF of the therewith created random process $Z[n]$.

In the present case $F_Y(\cdot)$ is a Gaussian CDF (i.e., complementary Q-function), since $Y[n]$ is normal distributed with zero mean and unit variance, which is guaranteed by the restrictions imposed on $Y[n]$ mentioned above. The polynomial $\mathfrak{p}_Y(y)$ shall resemble this percentile transformation procedure, $\mathfrak{p}_Y(y) \approx F_{Z,\text{target}}^{-1}(F_Y(y))$, thus, define the targeted distribution for the output process $Z[n]$. Proximity of the targeted CDF $F_{Z,\text{target}}(\cdot)$ and the actually realized CDF $F_Z(\cdot)$ is achieved by an ordinary polynomial curve fitting, for example, with the least-squares method [49]. Furthermore, Sec. V shows that for various types of distributions $F_{Z,\text{target}}(\cdot)$, a low-order polynomial approximation $p_Y(y)$ is satisfactory. The exact PDF $f_Z(z)$ of the output process $Z[n]$ can be computed by deploying the *fundamental theorem on transformation of random variables*, see [19, p. 130].

The input process $Y[n]$ exhibits non-trivial auto-correlations and cross-correlations, introduced by the preceding blocks (i.e., 2, 3). Those are influenced by the polynomial transformation. The auto-correlation function $\rho_{ZZ}[m]$ of the output process $Z[n]$ is a transformed version of the ACF $\rho_{YY}[m]$,

$$\rho_{ZZ}[m] = \mathfrak{p}_\rho(\rho_{YY}[m]), \tag{3}$$

see Annex I-A, where $\mathfrak{p}_\rho(\rho)$ is a polynomial, depending on the coefficients α_p of the polynomial $\mathfrak{p}_Y(y)$. Similarly, the cross-correlation function $\rho_{Z_iZ_l}[m]$ between the processes $Z_i[n]$ and $Z_l[n]$ is a transformed version of the XCF $\rho_{Y_iY_l}[m]$ between the processes $Y_i[n]$ and $Y_l[n]$,

$$\rho_{Z_iZ_l}[m] = \mathfrak{p}_{\rho,il}(\rho_{Y_iY_l}[m]), \tag{4}$$

see Annex I-B, where $\mathfrak{p}_{\rho,il}(\rho)$ is a polynomial, depending on the coefficients $\alpha_{p,i}$ of the polynomial $\mathfrak{p}_{Y,i}(y)$ and $\alpha_{p,l}$ of the polynomial $\mathfrak{p}_{Y,l}(y)$.

B. LINEAR TIME INVARIANT FILTER

The process $X[n]$ is passed through an LTI filter with real-valued impulse response $h[m]$, block 3. This filter fulfills the task of introducing auto-correlation to $X[n]$, resulting in the process

$$Y[n] = \sum_{m=-\infty}^{\infty} X[m] \cdot h[n-m] = X[n] * h[n], \tag{5}$$

where $*$ denotes the convolution operation. The reason for the combination of a Gaussian process with an LTI filter is the closure property of the set of all Gaussian processes on the addition operation and, especially, on linear combinations. It implies that any Gaussian random process $X[n]$ which is transformed by a linear filter $h[m]$ results again in a Gaussian process $Y[n]$. Thereby, the mean and variance of the output process are changed to [19, p. 398], $\mu_Y = \sum_{m=-\infty}^{\infty} h[m] \cdot \mu_X$ and $\sigma_Y^2 = \sum_{m=-\infty}^{\infty} (h[m])^2 \cdot \sigma_X^2$, where μ denotes the mean and σ^2 the variance. If the Gaussian input sequence $X[n]$ has zero mean and unit variance (which is one of the requirements mentioned above) and the sum of all squared filter coefficients $h[m]$ equals one, it is guaranteed that the distribution of the output sequence $Y[n]$ is also Gaussian with zero mean and unit variance and fulfills the restrictions on $Y[n]$. Hence, the closure property allows to introduce an ACF to the random process $X[n]$ by an arbitrary linear filter $h[m]$ without changing its distribution, provided it satisfies

$$\sigma_h^2 = \sum_{m=-\infty}^{\infty} (h[m])^2 \stackrel{!}{=} 1. \tag{6}$$

The ACF for a (wide-sense) stationary and ergodic identically distributed random process $Y[n]$ is thereby defined as

$$\begin{aligned} \rho_{YY}[m] &\doteq \frac{\gamma_{YY}[m] - \mu_Y^2}{\sigma_Y^2} \\ &= \frac{E\{(Y[n] - \mu_Y)(Y[n+m] - \mu_Y)\}}{\sigma_Y^2}, \end{aligned} \tag{7}$$

with the expectation operation $E\{\cdot\}$, the mean μ_Y, the variance σ_Y^2 and the *unnormalized* ACF $\gamma_{YY}[m]$. The XCF is defined similar, by exchanging the random process $Y[n]$ with two distinct processes $Y_i[n]$ and $Y_l[n]$, yielding $\rho_{Yi,Yl}[m]$. The ACF introduced by the LTI filter $h[m]$ to the process $Y[n]$ calculates to (see [19, p. 401])

$$\begin{aligned} \rho_{YY}[m] &= \frac{\gamma_{YY}[m]}{\sigma_Y^2} = \frac{\gamma_{hh}[m] * \gamma_{XX}[m]}{\sigma_h^2 \cdot \sigma_X^2} \\ &= \frac{\gamma_{hh}[m] * \delta[m]}{\sigma_h^2 \cdot 1} = \frac{\sigma_h^2 \cdot \rho_{hh}[m]}{\sigma_h^2} \\ &= \rho_{hh}[m] = h[m] * h[-m]. \end{aligned} \tag{8}$$

with the unit impulse sequence $\delta[m]$. The condition in Eq. (6) does not effect the auto-correlation function $\rho_{YY}[m]$, since it is normalized by the variance of the output process σ_h^2, as observed in the above equation. Therefore, any scaled version of the applied LTI filter results in the same autocorrelation function. Conversely, this means that Eq. (6) can always be satisfied by scaling any arbitrary $h[m]$ with a constant. This fact *decouples the problems of fitting CDF and ACF* to data, being responsible for a parsimonious and efficient treatment

of the overall fitting problem (the main reason for the choice of this model).

The linear filter comprises of two components, an Auto-Regressive (AR) component $\phi(B)$ and a Moving-Average (MA) component $\theta(B)$, which together constitute the ARMA model. The AR branch feeds a linear combination of the past output values $Y[n-m]$ back to the actual output value, the MA unit feeds a linear combination of the past input values $X[n-m]$ to the actual output $Y[n]$. By introducing the backshift operator B (i.e., $B\ X[n]=X[n-1]$), $\phi(B)$ and $\theta(B)$ can be interpreted as polynomials in B, where the power of B indicates how often a backshift is performed. The linear filter satisfies the difference equation (see [21, p. 8ff.])

$$\phi(B) \cdot Y[n] = \theta(B) \cdot X[n]. \tag{9}$$

Assessing the system behavior relies on the calculation of the impulse response $h[m]$ from the ARMA parameters $\phi(B)$ and $\theta(B)$. This can be achieved recursively by assuming $X[n]=\delta[n]$ and $h[n]=Y[n]$, starting from index $n=0$ and approaching $n\to\infty$. Besides, solving the difference equation for $Y[n]$, results in the polynomial $\psi(B)=\phi^{-1}(B)\theta(B) = \sum_{m=0}^{\infty} \psi_m B^m$, which directly leads to the impulse response by assigning $h[m]\doteq\psi_m$, $\forall\ 0\leq m\leq\infty$.

The linear filters $h_i[m]$ affect the cross-correlation function $\rho_{Y_iY_l}[m]$ between $Y_i[n]$ and $Y_l[n]$. In analogy to Eq. (8), we obtain

$$\begin{aligned}\rho_{Y_iY_l}[m] &= \rho_{X_iX_l}[m] * \rho_{h_ih_l}[m] \\ &= \rho_{X_iX_l}[m] * h_i[m] * h_l[-m].\end{aligned} \tag{10}$$

This equation allows for the analytical calculation of the transformation of the XCF induced by the introduction of ACFs to the random processes. Hence, alike Eq. (3) and Eq. (4), this equation is the key feature for the separation of the fitting problems of ACFs and XCFs.

C. WEIGHTING MATRIX

The matrix $\mathbf{G}[m]$, block 2, serves the purpose of introducing cross-correlations into the processes $X_i[n]$. It combines the processes $W_j[n]$ by weighted addition. However, the elements $g_{ij}[m]$ of matrix $\mathbf{G}[m]$ are sequences of weights in the timing lag m, in the most general case. This allows for the interpretation of each $g_{ij}[m]$ as the impulse response of a linear filter or, equivalently, as polynomial $g_{ij}(B)$ in the backshift operator B. Thus, matrix $\mathbf{G}[m]$ is equivalent to a matrix polynomial $\mathbf{G}(B)$ in B (see [21, p. 551ff.] and [31, p. 401ff.]). The input-output relation can be conveniently written in matrix notation as

$$\mathbf{X}[n] = \mathbf{G}(B) \cdot \mathbf{W}[n], \tag{11a}$$

where $\mathbf{X}[n]$ and $\mathbf{W}[n]$ are vector valued random processes composed by all $X_i[n]$ and $W_j[n]$. On the other hand, the element-wise output relation can be written as

$$X_i[n] = \sum_{j=1}^{J} g_{ij}[n] * W_j[n], \tag{11b}$$

in which each element $g_{ij}[m]$ denotes a linear filter in m.

The cross-correlations introduced by matrix $\mathbf{G}[m]$ can be calculated by deploying the backshift notation $\mathbf{G}(B)$, namely,

$$\Gamma_X(B) = \mathbf{G}(B) \cdot \mathbf{G}^T(B^{-1}), \tag{12a}$$

where $(\cdot)^T$ denotes the transposed of the matrix. The corresponding matrix in the *time lag* domain is denoted by $\Gamma_X[m]$, with each element $\gamma_{X_iX_l}[m]$ being the specific XCF between the respective random processes $X_i[n]$ and $X_l[n]$. These elements calculate to

$$\rho_{X_iX_l}[m] = \gamma_{X_iX_l}[m] = \sum_{j=1}^{J} g_{ij}[m] * g_{lj}[-m], \tag{12b}$$

where $\gamma_{X_iX_l}[m]=\rho_{X_iX_l}[m]$ due to the normalization postulated by Eq. (13).

As already mentioned, $\mathbf{G}[m]$ is restricted to the set of matrices which fulfill the following conditions for the output processes: (i) all $X_i[n]$ must be Gaussian distributed with zero mean and unit variance and (ii) all $X_i[n]$ must have zero auto-correlation for lags $m\neq 0$.

The first condition requires that the squared sum of all row elements $g_{ij}[m]$ of $\mathbf{G}[m]$ equals one for all rows i,

$$\sum_{j=1}^{J} \sum_{m=-\infty}^{\infty} (g_{ij}[m])^2 \stackrel{!}{=} 1. \tag{13}$$

Due to the closure property of the Gaussian distribution on linear combinations, Gaussianity as well as the zero mean are preserved for $X_i[n]$. A squared sum equal to one, see Eq. (13), ensures that the variance of $X_i[n]$ equals one; hence, the first condition is fulfilled.

The second condition (i.e., zero ACF for all lags unequal to zero) is equivalent to forcing all diagonal elements of $\Gamma_X[m]$ to

$$\rho_{X_iX_i}[m] \stackrel{!}{=} 1 \cdot B^0 = 1. \tag{14}$$

It must be ensured by the respective fitting procedure (see Sec. IV-C). The condition guarantees that the ACFs of the output processes are independent of the matrix $\mathbf{G}[m]$. This seems to be an overhead, since $\mathbf{G}[m]$ could also introduce an ACF to the processes and, thereby, incorporate the linear filter $h[m]$. The reason of the separation of the two blocks is the possibility of different targeted fitting accuracies for ACF and XCF. Specific types of data traffic may, for example, require that the ACF is modeled accurately up to a lag of 10^4, whereas it is considered as sufficient to model the XCF only at lag 0. This task is simplified by splitting both fitting problems into two independent sub-problems. Further, fitted models tend to have less parameters in this case.

An important class of matrices which satisfies this condition is the set of real valued matrices $\mathbf{G}$ without any backshift operation. Such matrices only define the cross-correlation coefficients $\rho_{X_iX_l}[0]$ between the processes $X_i[n]$ and $X_l[n]$ and do not introduce cross-correlations at any other lag. This is sufficient for many practical applications (see Sec. V);

the input-output relation Eq. (11) reduces to an ordinary matrix multiplication $\mathbf{X}[n]=\mathbf{G}\mathbf{W}[n]$.

D. NORMAL I.I.D. PROCESSES

The proposed traffic generation method requires J i.i.d. Gaussian random processes $W_j[n]$ with zero mean and unit variance, block 1. This is convenient for simulation purposes, since most modern computer systems provide predefined, computationally efficient routines for the generation of high-quality normal distributed random variables [20]. Furthermore, all J processes must be independent, hence, can be interpreted as a J-dimensional i.i.d. random process. The generation of such processes is feasible up to high dimensionality [50]. Thus, all three requirements for the intermediate process $W_j[n]$ imposed at the beginning of this section are fulfilled. The number of processes J is determined by the number of output processes I and the desired structure of interdependencies (i.e., XCFs). The exact number is determined during the fitting process, see Sec. IV.

IV. BUILDING TARMA MODELS FROM RECORDED TRAFFIC

Procedures for fitting TARMA models to measurement data are illustrated in the following. Thereby, well established methods for fitting ARMA processes and polynomial regression are partly reused. Scripts performing the fitting process fully automatic can be downloaded from [51]. The fitting process has to be performed beginning with the polynomial and proceeding in reverse order, from block 4 to 2, see Fig. 2. By doing so the fitting problems for each block are decoupled. Nevertheless, the fitting procedure for each component has to account for the influences of the consecutive components on the respective statistical measure. For example, the ACF which is introduced by the linear filter $h[m]$ is altered by the polynomial $\mathfrak{p}_Y(y)$. The influences can be assessed analytically by the functions presented in Sec. III, e.g., Eq. (3), Eq. (4) and Eq. (10). This is one of the big advantages of the proposed model. In the following we describe the fitting processes for each block separately. Thereby the targeted quantities of the resulting model are denoted by the subscript $(\cdot)_{\text{target}}$; those are, for example, obtained from traced data traffic or from analytical models.

A. POLYNOMIAL TRANSFORMATION

The first block to be considered is the polynomial transformation $\mathfrak{p}_Y(\cdot)$, block 4. It shall introduce an arbitrary CDF to $Z[n]$. As already stated in Sec. III-A, $\mathfrak{p}_Y(\cdot)$ shall approximate a quantile-transformation procedure, confer Eq. (2). This can be achieved by solving a least-squares fitting problem [49]. Thereby the sample points to be fit by polynomial regression are pairs of ω_k-quantiles $(\Omega_Z, \Omega_Y)_k$ from the Gaussian CDF $F_Y(\cdot)$ of $Y[n]$ and the targeted CDF $F_{Z,\text{target}}(\cdot)$ of $Z[n]$, namely,

$$(\Omega_Z, \Omega_Y)_k = \left(F^{-1}_{Z,\text{target}}(\omega_k), F^{-1}_{Y}(\omega_k)\right) \qquad 0 < \omega_k < 1.$$

The sample points $(\Omega_Z, \Omega_Y)_k$ can be arranged in a Q-Q-plot. An illustration of the determination of sample points is given in Fig. 3, wherein the process $Z[n]$ has uniform distribution. The number of quantile values ω_k for the polynomial regression, as well as their position and spacing is an open point for optimization; hence, depending on the designers needs. For the rest of this work equidistant spacing from zero to one is assumed, $0 < \omega_k < 1$, excluding both limiting values, since they would yield $(\Omega_Z, \Omega_Y)=(\Omega_Z, \pm\infty)$ and are not suited for a polynomial regression. It is irrelevant if the quantile values either stem from measurements or a certain type of analytical distribution.

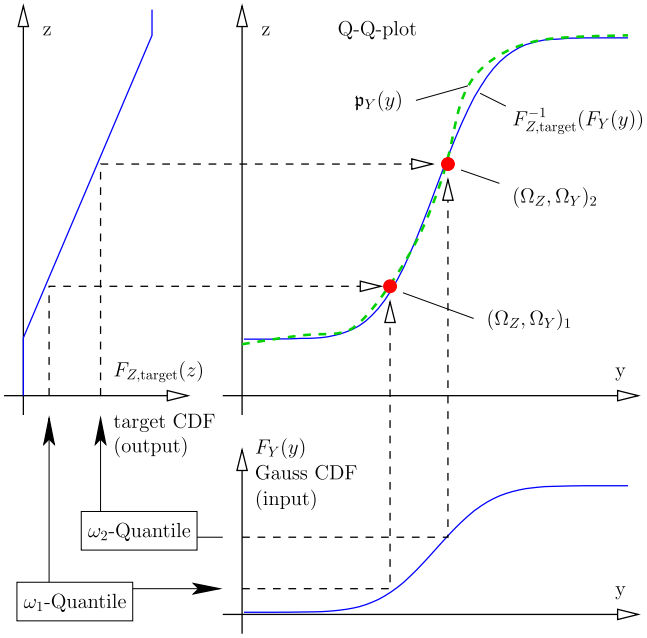

FIGURE 3. Q-Q-plot for obtaining the sample points $(\Omega_Z, \Omega_Y)_k$ to which the polynomial $\mathfrak{p}_Y(y)$ must be fitted.

The order P of the polynomial plays an important role for the quality of the fit. If the order is high enough, it is possible to fit any number of points with arbitrary accuracy; however, polynomial fitting has poor interpolation properties, i.e., the fit tends to oscillate between points (over-fitting). We recommend to use low-order polynomials. Furthermore, the computational complexity for the generation of random samples is strongly reduced for small P. A comparison of the quality of fit for different polynomial orders is given in Sec. V.

B. LINEAR FILTERING

Block 3, the linear filter $h[m]$, shall introduce an ACF to $Z[n]$ by introducing a corresponding ACF to $Y[n]$. In literature two approaches for ARMA modeling are prevalent: (i) ACF based approaches (e.g., Yule-Walker equations, Power Spectral Density (PSD) based approaches), which require $\rho_{YY,\text{target}}[m]$ as input and (ii) direct methods based on the data itself (e.g., Maximum Likelihood (ML) modeling), requiring $Y_{\text{target}}[n]$ as input. It has to be taken into

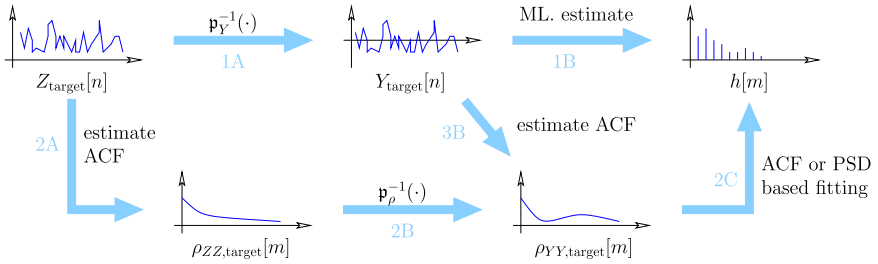

FIGURE 4. Possible approaches to fit a linear filter (ARMA model) to traced data.

account that the polynomial transformation $\mathfrak{p}_Y(y)$, block 4, influences the ACF of the output process $Z[n]$ according to Eq. (3). According to which fitting method shall be applied, the respective input quantity has to be pre-distorted such that the influence of the polynomial transformation is taken into account. A respective graphical representation is given in Fig. 4. Thus, either

- the input data for fitting has to be manipulated, according to $Y_{\text{target}}[n]{=}\mathfrak{p}_Y^{-1}(Z_{\text{target}}[n])$ (see Fig. 4: Step 1A) or
- the input ACF for fitting has to be manipulated, according to $\rho_{YY,\text{target}}[m]{=}\mathfrak{p}_\rho^{-1}(\rho_{ZZ,\text{target}}[m])$ (see Fig. 4: Step 2B).

Both of these pre-distortion methods require the inversion of a polynomial, or, equivalently, finding the respective roots, which is analytically not feasible for any $P{>}4$. However, this frequent problem can efficiently be solved numerically with high accuracy. The polynomial $\mathfrak{p}_\rho^{-1}(\cdot)$ is usually smoother than the polynomial $\mathfrak{p}_Y^{-1}(\cdot)$ and allows for a unique solution of the inversion problem. Therefore, the fitting procedure involving the transformation of the ACF (see Fig. 4: Step 2A–2B–2C) yields most probably better results than the direct fitting approach involving the transformation of the data (see Fig. 4: Step 1A–1B or 1A–3B–2C). A counterexample is modeling of video sequences, see Sec. VI-C.

Having obtained either $\rho_{YY,\text{target}}[m]$ or $Y_{\text{target}}[n]$ as input for the respective fitting procedure for the linear filter, any arbitrary ARMA modeling approach can be applied for computing the ARMA parameters $\phi(B)$ and $\theta(B)$. Those are numerous in literature [21] and [31], including various software solutions.

A typical property of ACFs of network traffic is Long Range Dependence (LRD). It is encountered for time series of various quantities of data traffic, such as packet-sizes, flow durations, packet counts and IATs [53]. LRDs have an impact on the queueing performance and are therefore important to be captured. However, several ARMA modeling procedures have problems to capture these effects. In [52] a method is presented which overcomes these problems: a variant of ACF-based fitting. It yields a parsimonious ARMA model with finite length and is thus only an approximation to an LRD process. Nevertheless, the fitting accuracy is high for any finite lag of the ACF and the generation of samples exhibits very low complexity. Prominent alternatives are the well-known Auto-Regressive Fractionally Integrated Moving-Average (ARFIMA) models [21, p. 428ff.], [54], yielding long-range dependent processes by fractional integration (summation). This can be translated to an equivalent ARMA model of (formally) infinite length, for which the traffic synthesis is computationally more expensive than for ordinary ARMA processes. In [42] *circulant embedding* is proposed. This method models the spectral properties of the targeted time series. Accordingly, synthetic traffic is generated in the spectral domain and mapped to the time domain by a Fourier transform; yielding higher complexity than ordinary ARMA processes. ACF and XCF are thereby captured simultaneously, the resulting model is however not parsimonious. A comparison of the feasible fitting methods is presented in Table 2.

TABLE 2. Comparison of fitting strategies for $h[m]$.

	ML	[52]	ARFIMA	circ. emb. [42]
$\mathfrak{p}^{-1}(\cdot)$ problems	✓			
LRD feasible		✓	✓	✓
Parsimonious	✓	✓	✓	
Gen. complex.	$\mathcal{O}(N)$	$\mathcal{O}(N)$	$\mathcal{O}(N\log(N))$	$\mathcal{O}(N\log(N))$

Finally, in order to suffice the restriction on $Y[n]$ (i.e., normally distributed with zero mean and unit variance), it has to be guaranteed that Eq. (6) is satisfied. This can be achieved by scaling $\theta(B)$ with a constant.

C. WEIGHTING MATRIX

The last block to be considered for the fitting procedure is block 2, the weighting matrix $\mathbf{G}[m]$. This block shall introduce XCFs between the I different output processes $Z_i[n]$, by introducing respective XCFs to $X_i[n]$. Again the influences from block 3 and 4 on the XCFs between the output processes have to be considered first; confer Eq. (10) and Eq. (4). In analogy to the fitting problem for the linear filter it is possible to either (i) fit $\mathbf{G}[m]$ to the random processes $X_{i,\text{target}}[n]$ or (ii) fit $\mathbf{G}[m]$ to all the XCFs $\rho_{X_iX_l,\text{target}}[m]$. How to obtain one of the above quantities is outlined in Fig. 5.

The first procedure is accomplished by (see Fig. 5: 1A–1B–1C):

- inverting the polynomial transformation (1A), $Y_{i,\text{target}}[n]{=}\mathfrak{p}_Y^{-1}(Z_{i,\text{target}})$,

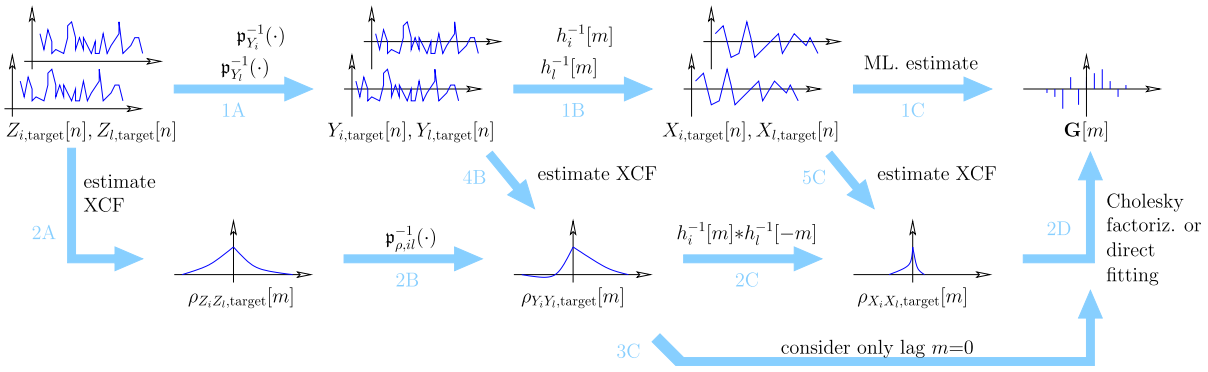

FIGURE 5. **Possible approaches to fit a polynomial matrix $\mathbf{G}[m]$ to traced data.**

- whitening the obtained sequence by applying $X_{i,\text{target}}[n]=Y_{i,\text{target}}[n]*h_i^{-1}[n]$ (1B) and
- applying a ML fitting approach (1C).

The second procedure comes in several flavors (see Fig. 5). For example, option 2A–2B–2C–2D can be performed as follows:

- calculate the XCFs of $Z[n]$ (2A),
- apply the inverse polynomial from Eq. (4) to it (2B), $\rho_{Y_iY_l,\text{target}}[m]=\mathfrak{p}_{\rho,il}^{-1}(\rho_{Z_iZ_l,\text{target}}[m])$,
- whiten the XCFs of $Y[n]$ (2C) by the inverse filters $h_i^{-1}[m]$ of Eq. (10), $\rho_{X_iX_l,\text{target}}[m]=\rho_{Y_iY_l,\text{target}}[m] * h_i^{-1}[m]*h_l^{-1}[-m]$ and
- calculate $\mathbf{G}[m]$ from Eq. (12) using the *Cholesky decomposition* (2D).

TABLE 3. **Comparison of fitting strategies for $\mathbf{G}[m]$.**

	Cholesky	direct	lag zero
Number of output processes I	any	2	any
Number of lags M	any	any	0
Number of input processes J	$(2M+1)I$	$(M+1)I$	I
Number of parameters	$\frac{1}{2}J(J-1)$	$2J$	$\frac{1}{2}I(I-1)$
Parsimonious			✓

As already mentioned in the context of ACF modeling, step 1A for inversion of the polynomials $\mathfrak{p}_{Y_i}^{-1}(\cdot)$ is problematic and shall be avoided. Fitting sequences which include it (e.g., 1A–1B–1C, see Fig. 5) are therefore unfavorable and yield alternative sequences (e.g., 2A–2B–2C–2D or 2A–2B–3C, see Fig. 5) more convenient for practical use.

Consequently, the central fitting step corresponds to either Step 2D or Step 3C. Two methods are available for this purpose: (i) the Cholesky-factorization or (ii) the direct method (only applicable to the case of two output processes). Unfortunately, both methods are not parsimonious (in contrast to the modeling approaches for PDFs and ACFs). They yield roughly one model parameter per lag for each XCF; consequently, the number of parameters becomes easily prohibitively large. Therefore we suggest to model only up to a few lags of the XCF (e.g., only lag 0), especially if more than two output processes shall be characterized. The fitting algorithms are described in Annex II. A comparison of the different fitting strategies is provided in Table 3. An evaluation of the impact of the number of considered lags on the model accuracy is given in Sec. V.

V. EVALUATION: CONCEPTUAL REMARKS

We evaluate the proposed modeling approach and analyze its capability of handling real network traffic with acceptable model complexity. Up to now we focused on the two most important properties of the proposed method: separability of the fitting problems and closed form analytical tractability. In the following further aspects are commented on, being of general interest in the context of source traffic modeling.

A. GENERAL REMARKS

1) PARSIMONIOUSNESS

A low number of model parameters is mostly desired for models, since it facilitates the reproducibility of fits and makes them less error-prone. TARMA models achieve this due to: (i) the presented fitting methods allow for parsimonious fitting of each of the three statistical measures individually (i.e., CDF, ACF and XCF) and (ii) the separability of the fitting problem guarantees independence between the number of parameters used for each measure. For example, if a high number of parameters is required for fitting the CDF with satisfactory accuracy, this has no influence on the number of parameters required for fitting the ACF. Modeling approaches for which the fitting problem is not separable suffer from the coupling of the number of parameters (e.g., variants of Markovian models).

2) EFFICIENT SAMPLE GENERATION

The computationally efficient generation of samples is guaranteed for TARMA processes. The reasons are (i) normal i.i.d. samples are efficiently generated by various known methods [20], [50], (ii) the weighting matrix requires J multiplications and additions per sample, (iii) the ARMA(P,Q) filter requires $P+Q$ multiplications and additions and (iv) the polynomial

transformation of order P requires $2P$ multiplications and $P+1$ additions. Summing up, each sample requires some tens of multiplications and additions, which allows for the generation of millions of samples per second on commodity hardware. An implementation in *Matlab* can be downloaded at [51].

3) HIGHER ORDER STATISTICS

Recent literature on traffic modeling addresses higher order statistics [16], which is not the focus of the present work. In [42] the authors discuss on possibilities how to fit respective quantities with TARMA models. Accordingly, this can be achieved by permuting the quantiles during the CDF modeling procedure (see Sec. IV-A). However, it remains unclear to which extend the uniqueness of the invertibility of the polynomials $\mathfrak{p}_{Y,i}(\cdot)$ and $\mathfrak{p}_{\rho,il}(\cdot)$ is influenced by the permutation procedure. Furthermore, the impact of various statistical measures on the modeling quality is unclear. Such measures are, for example, *bi-spectra* or *joint distribution functions* (i.e., only partially marginalized). Future work has to clarify on this issue, possibly by providing a ranking of the most important statistical measures in the context of traffic modeling.

B. REMARKS ON THE DISTRIBUTION

The distribution of the output processes $Z_i[n]$ is generated by the polynomial $\mathfrak{p}_Y(y)$, hence, it is a non-parametric distribution. The question arises of how accurately standard distributions can be represented. Of course the Gaussian distribution, as well as some other distributions (e.g., chi-squared distribution) can be perfectly resembled by a non-parametric system, since they are a polynomial transformations of Gaussian random processes. An evaluation of other standard distributions is shown in Fig. 6 (a), where the maximum CDF distance over the polynomial order P is depicted. The log-normal, uniform, exponential, Weibull and gamma distributions are presented. As expected, a higher polynomial order leads to better fitting accuracy. Remarkably low errors are achievable for polynomials with moderate order, say P=10, which is the key feature for a parsimonious representation. The uniform and exponential distributions tend to slightly worse accuracies than other distributions, due to the point(s) of discontinuity of the respective PDFs. Further, the error-floor at roughly 10^{-5} results from the choice of percentile points $(\Omega_Z, \Omega_Y)_i$ to which the polynomials have been fitted. In the present case the first and last points correspond to the 10^{-5} and $(1-10^{-5})$ percentiles, hence, beyond those values the congruence of both CDFs is not guaranteed.

Whenever rare events are simulated (e.g., packet loss, bit errors), it is crucial that the tail of the distribution is accurately modeled, since such events are often caused by respective random samples. Hence, a maximum CDF distance of 10^{-5} may not be tolerable in the respective region. For example, by assessing connection time-outs it is important to accurately model the rare events with IATs of up to some seconds,

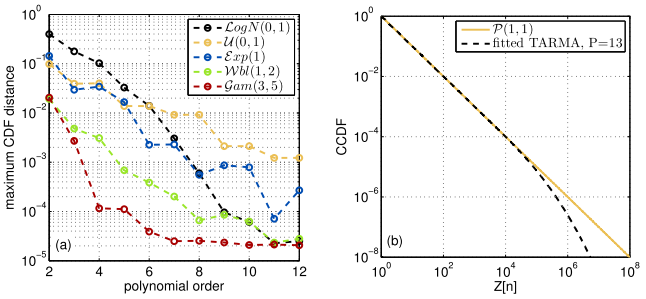

FIGURE 6. (a): Fitting quality of transformed Gaussian distributions over the polynomial order *P* of $\mathfrak{p}_Y(y)$ for log-normal, uniform, exponential, Weibull and gamma distributions; parameters adhere to the *Matlab* syntax. (b): Survival function of a Pareto $\mathcal{P}(1,1)$ distributed variable and the respective fit of a transformed Gaussian.

whereas the body of the distribution with IATs in the order of milliseconds is of minor interest. In such cases, it is recommended to emphasize the region of interest by a higher density of quantile points $(\Omega_Z, \Omega_Y)_i$ and to sacrifice some accuracy in other regions. An example is given in Fig. 6 (b), where the survival function of a Pareto distribution, $\mathcal{P}(1,1)$, is compared to the respective model fit. Due to the increased number of quantile points in the tail deployed for fitting, an acceptable fitting accuracy is obtained even for quantiles of up to $(1-10^{-8})$. Consequently, truncated heavy-tailed distributions can be reproduced well by our model.

Another very common property associated with network traffic is one-sided positive distributions. Gaussian random processes, however, have a domain of $[-\infty, \infty]$. Thus, the polynomial transformation should guarantee that the probability of negative values of $Z[n]$ equals zero. Due to the poor extrapolation properties of polynomials this is hardly achievable in practice. This means that in the remote case of a Gaussian sample of $Y[n]$ being close to $-\infty$ negative values of $Z[n]$ may occur. In order to absolutely prevent such cases, the values of the samples $Z[n]$ shall be limited to a minimum of zero. Theoretically, this is another non-linear transform introduced to $Y[n]$ which changes the ACFs and XCFs of $Z_i[n]$; nevertheless, due to the very low probability of occurrence of negative samples, these changes may be neglected.

Finally, network traffic may exhibit mixed continuous and discrete distributions. For example, the PS may be modeled well by a continuous distribution but exhibits a discrete number of peaks at certain common packet sizes. Such peaks are caused by routines in protocols (e.g., TCP acknowledgments). The presented modeling approach is not suited for representing single peaks, but interpolates between peaks. This behavior can be observed in Fig. 8 (a) and Fig. 9(a). The advantage is that peaks resulting from a small sample size are smoothed [see Fig. 9(a)] but, on the other hand, also peaks with a concrete physical interpretation are attenuated [see Fig. 8 (a)]. For applications which rely on the accurate representation of a limited number of peaks, it is therefore recommended to use a different modeling approach, such as Markovian models.

C. REMARKS ON THE AUTO-CORRELATION FUNCTION

The ACF of the process $Z[n]$ is, according to Eq. (3), a transformed version of the auto-correlation function of $Y[n]$. Thereby, the absolute value of the transformed ACF is always smaller or equal to the absolute value of the original ACF, see [43, p. 133, Lemma 7.1]. Since all ACFs are defined on the interval $\mathbb{A}=]-1, 1]$, both domain and codomain of the function $\mathfrak{p}_\rho(\cdot)$ equal $\mathbb{A}$, $\mathfrak{p}_\rho : \mathbb{A}\rightarrow\mathbb{A}$. However, if the image of $\mathfrak{p}_\rho(\cdot)$ in its codomain is a subset of $\mathbb{A}$, certain values of the desired ACF may not be realizable with any kind of LTI filter. Since both 1 and 0 are by definition contained by the image of $\mathfrak{p}_\rho(\cdot)$, all positive values are realizable for the ACF of $Z[n]$. Negative values on the other hand are not, whereas empirical trials suggest an increase of skewness of the CDF of $Z[n]$ to narrow the image of $\mathfrak{p}_\rho(\cdot)$ in its codomain (see [42]).

Long range dependencies are another typical property of network traffic [8], [29], [55]. Our modeling approach is theoretically not able to reproduce long-range dependence, since this property is equivalent to $\sum_{m=1}^{\infty}|\rho_{ZZ}[m]|=\infty$ [56], which is a contradiction to the requirement in Eq. (6). Nevertheless, it is possible to introduce dependencies which are arbitrary long (but less than ∞) by using ARMA(1,0) model or higher. An example is given in Fig. 8 (c), where dependencies up to lag $m=10^4$ are modeled according to [52] with high accuracy. Furthermore, also other authors successfully approximated long-range dependencies by deploying short memory models, for example, Markovian models [34] and TES models [37].

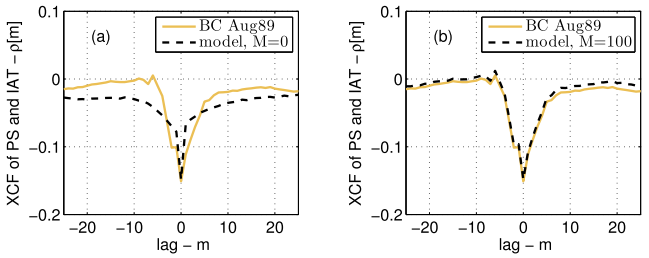

FIGURE 7. Fitting the XCF of PS and IAT of the *Bellcore Aug89* data set, (a): only for lag $M=0$, without whitening, (b): up to lag $M=100$, Cholesky decomposition.

D. REMARKS ON THE CROSS-CORRELATION FUNCTIONS

The parsimoniousness of the matrix $\mathbf{G}(B)$ strongly depends on the amount of output processes I and the maximum lag M up to which the XCFs shall be modeled. This requires to keep the value M small. The modeling accuracy, on the other hand, suffers from small values of M. This effect can be observed in Fig. 7, where the XCF between PS and IAT of the *Bellcore Aug89* trace [57] is modeled. The Fig. 7 (b) compares the real trace and its synthetic counterpart, where a maximum lag of $M=100$ is considered. In this case the modeling accuracy is very high, however a total of $J=202$ input processes $W_j[n]$ are required, with $\mathbf{G}(B)$ having more than 20 000 parameters. Considering only small values of M yields inaccuracies in general, which is due to the linear filters $h_i[m]$. They spread the model error of $\mathbf{G}(B)$ made for lags $|m|>M$ (concerning $X_i[n]$) over the whole range of m (concerning $Y_i[n]$). Fitting only lag zero (i.e., $M=0$) without performing the whitening operation (i.e., using $\rho_{Y_i,Y_l,\text{target}}[m]$ instead of $\rho_{X_i,X_l,\text{target}}[m]$ as input for fitting) constitutes a remedy to this problem. In this case the target XCF can perfectly be reached at lag zero, whereas errors at all other lags have to be accepted, see Fig. 7 (a). This approach is preferable compared to small values of M (but $M\neq0$), where whitening is required. Future work has to target the problem of expressing $\mathbf{G}(B)$ in a parsimonious way for large lags M.

TABLE 4. Performance evaluation, maximum distances.

	$F_{\text{PS}}(z)$	$F_{\text{IAT}}(z)$	$\rho_{\text{PS,PS}}[m]$	$\rho_{\text{IAT,IAT}}[m]$
Bellcore Aug89	0.2330	0.0064	0.0303	0.0239
openarena	0.0142	/	0.0626	/
Lord of the Rings I	0.0136	/	0.0491	/

VI. DEPLOYMENT EXAMPLE: MODELING RECORDED NETWORK TRAFFIC

To demonstrate the real-world performance of the proposed approach, TARMA models are presented for traced source traffic. Thereby, three different traces are fitted, in order to demonstrate the generality of this approach. They cover (i) the popular *Bellcore Aug89* data set [8]. [57], (ii) traced traffic from the online game *openarena* [58] and (iii) the online available MPEG-4 trace of the movie *Lord of the Rings I* [59]. Beside of evaluating the quality-of-fit by assessing the congruence of the three statistical measures (i.e., CDF, ACF, XCF), see Table 4, a benchmark is provided by feeding the traced traffic as well as respective emulated traffic to a G/G/1 queue.

A. AGGREGATED DATA TRAFFIC

The *Bellcore Aug89* data set [8], [57] is not typical source traffic but rather aggregated traffic, however, often used as reference for traffic modeling approaches [34], [45]. A fitted TARMA processes is presented in Fig. 8, where PS and IAT have been modeled.The two leftmost figures present the ECDFs of PS and IAT, respectively. Thereby the polynomial order of the fitted transformation are both equal to $P=5$. It is clearly visible that the discrete steps of the PS are smoothed by the model, resulting in a relatively large maximum distance, see Table 4. The ECDF of the IAT on the other hand is modeled well over two decades. The fits of the ACFs of the IAT and the respective ARMA(5,5) fits, are presented in Fig. 8 (c); they exhibit good accuracy over four decades. The XCF between PS and IAT has been modeled for several maximum lags M; shown in Sec. V, Fig. 7. The respective value for $m=0$ is negative, which is intuitively explainable by the fact that big PSs are likely to be followed by short IATs due to packet fragmentation. Fig. 8 (d) shows the survival function of the queuing response of the recorded and modeled traffic for different utilizations (i.e., 20%, 50% and 80%), yielding congruency for all three cases.

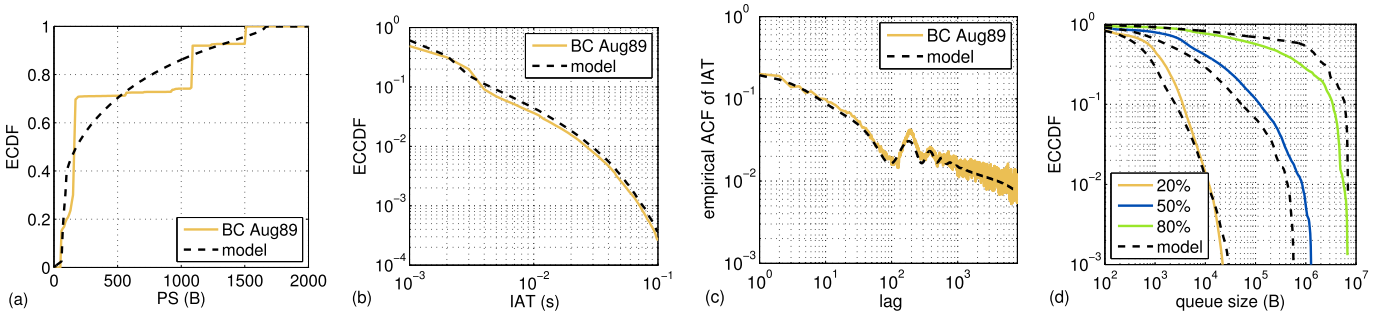

FIGURE 8. Fitting the *Bellcore Aug89* data set, (a): ECDFs of the PS, note that the discrete steps are smoothed by the modeling approach, (b): survival function of the IAT, (c): ACF of the IAT, (d): queueing response of the traffic for various utilizations.

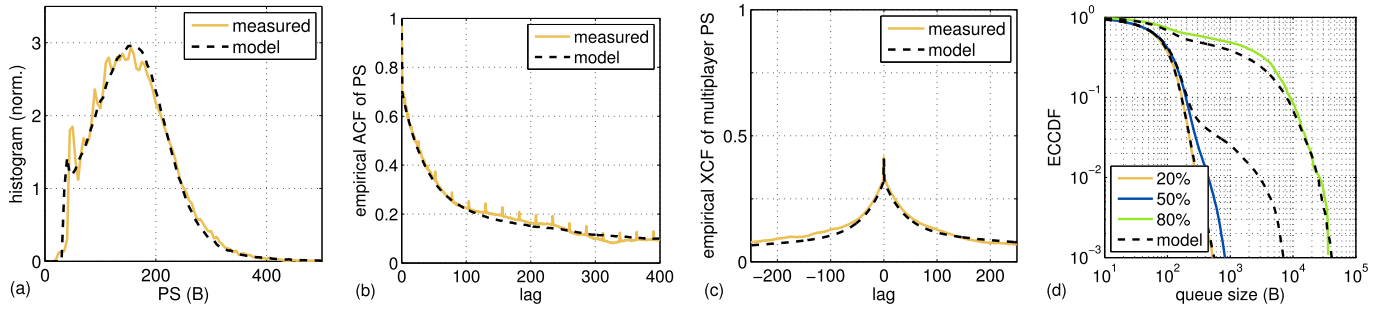

FIGURE 9. Fitting the *openarena* data set, (a): normalized histogram of the PS, (b): ACF of the PS, (c): XCF between PS processes of multiple players, (d): queueing response for different utilizations.

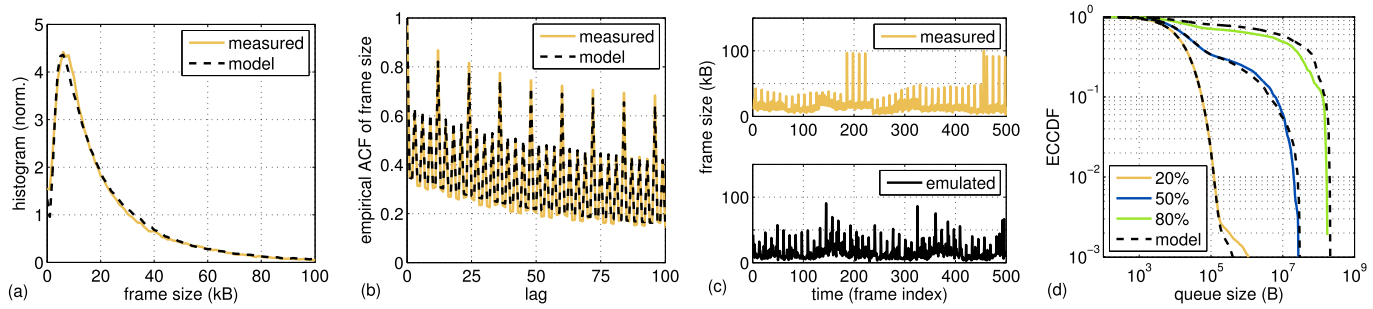

FIGURE 10. Fitting the *Lord of the Rings I* data set, (a): normalized histogram of the frame size, interleaved I,P and B-frames, (b): ACF of the frame size of interleaved process, (c): time series of the frame size, (d): queueing response.

B. ONLINE GAMING TRAFFIC

For obtaining the *openarena* [58] data set, we sniffed IP packets in the downlink direction at a dedicated game server, which was serving two players. We observed four sessions of 10 min each, with a total of roughly 50 000 packets per player. Since the packet IAT was constant with 40 ms, only the PS is modeled, however, jointly for both players.The PDF fit is evaluated in Fig. 9(a), where the polynomial order equals P=5. The fitted ACF is shown in Fig. 9(a), where an ARMA(5,5) model is deployed. The XCF is modeled only for the lag m=0, as described in Sec. IV, yielding a strong positive cross-correlation, see Fig. 9(c). Note that the XCFs is not only congruent at lag m=0, but also at all other lags up to 250. The reason is that the XCF is altered by the linear filter according to Eq. (10); thus, fitting the ACF already ensures that also the modeled XCF is close to its recorded counterpart. The evaluation of the queueing performance in Fig. 9(d) shows, that model and real data perform very similar, except for medium utilization U=50%. This is possibly caused by the brevity of the data set, provoking inaccuracies in the estimation of the ACF at high lags and the queue length itself.

C. VIDEO STREAMING TRAFFIC

The third traffic type is MPEG-4 video traffic, whereas the online available trace of the movie *Lord of the Rings I* [59] was considered. MPEG-4 videos consist of Group of Pictures

(GOPs), each of which composed of a combination of three different frame types (i.e., I, P and B-frames). In the present case the GOP exhibits a size of 12 frames according to the following structure: IBBPBBPBBPBB. For capturing this structure we modeled each frame-type as separate stream and introduced strong cross-correlation between them. The output stream was composed by interleaving the respective single streams according to the above mentioned GOP structure.

Thereby, all ACFs have been equal and the cross-correlation coefficients were one ($\rho_{IP}[0]=\rho_{PB}[0]=\rho_{BI}[0]=1$), only the distributions were changed from frame-type to frame-type. This approach is common in literature [3]; in the present context it can be interpreted as seasonal ARMA model [21, p. 353ff.]. It overcomes the problem of missing observations in the streams for single frame-types during the fitting procedure. For example, samples of I-frames can only be observed once in an entire GOP (e.g., eleven frames missing, one frame present). By assuming all correlation coefficients to one, it is legitimate to construct *one* background stream from the three streams of frame-types by pre-distortion, see Fig. 4, Step 1A. This stream has no missing observations and is Gaussian with zero mean and unit variance; it can directly be deployed for the derivation of $h[m]$. Therefore, the fitting procedure for the ACF shall rather follow 1A–1B or 1A–3B–2C than 2A–2B–2C, see Fig. 4.

An evaluation of the interleaved output stream is shown in Fig. 10 and Tab. 4. The leftmost plot, Fig. 10 (a), shows the PDF which is a superposition of the PDFs of the individual frame-types. The ACF, shown in Fig. 10 (b), is very peaky due to the interleaving described above. A visual comparison of the recorded and synthetic data streams is given in Fig. 10 (c). The queueing performances are compared in Fig. 10 (d). All plots exhibit a convincing quality-of-fit, see Table 4.

VII. CONCLUSION

We address the problem of designing a generative model for arbitrary network source traffic, with focus on multivariate stationary random processes. They emulate certain physical quantities of the measured traffic, for example, IP packet-size or packet inter-arrival time. For each random process three statistical measures are considered, namely, the distribution, the auto-correlation function and the cross-correlation function with other processes. All of them are known for their strong influence on the network behavior.

We propose a modeling approach based on Transformed Auto-Regressive Moving-Average (TARMA) processes. This approach allows for decoupling the overall modeling problem into three independent sub-problems, one for each statistical measure. Thereby, each problem is solvable by standard techniques. The decoupling is enabled by the structure of the model, consisting of four entities: (i) a random number generator which produces normal i.i.d. random processes, (ii) a polynomial weighting matrix, introducing cross-correlations to the processes, (iii) LTI filters which introduce arbitrary auto-correlations and (iv) memoryless polynomial non-linearities which transform the Gaussian random samples to arbitrary distributions. The analytical derivations for all relevant statistical measures is feasible (see Annex I), which is crucial for efficient model fitting.

Advantages of this method are its complete analytical tractability, parsimoniousness in the number of model parameters, the fitting procedure deploys only efficient standard techniques and the generation of samples exhibits low complexity.

Exemplary models for different traffic types are provided, which expose the generality of this approach. Online-gaming traffic is modeled, where packet size and packet inter-arrival times are emulated, as well as cross-correlations between multiple players. Further, a model for video traffic is shown, where the frame-size processes are emulated and combined to a single video streaming process. Besides the applicability of the approach to network source traffic, we also indicate its usefulness for aggregated network traffic by modeling the well known *Bellcore Aug89* trace. Thereby, the packet-size and packet inter-arrival time are considered as cross-correlated random processes. In order to evaluate the proposed method by an unrelated statistical measure, the traffic traces as well as synthetic traffic were fed to a single-server queue (G/G/1 queue). The resulting queue responses show excellent congruency in all evaluated cases.

ACKNOWLEDGMENT

This work was supported by the *EU FP7 LOLA* project (www.ict-lola.eu) and the DARWIN 4 FFG–Comet project (www.ftw.at).

APPENDIX I. TRANSFORMATION FORMULAS

A. ACF TRANSFORMATION

Let $Y[n]$ denote a Gaussian random process with zero mean, unit variance and auto–correlation function $\rho_{YY}[m]$ and $Z[n]$ the random process obtained by the transformation of $Y[n]$ by a polynomial $\mathfrak{p}_Y(\cdot)$ according to $Z[n] = \mathfrak{p}_Y(Y[n]) = \sum_{p=0}^{P} \alpha_p \cdot (Y[n])^p$. Then the auto–correlation function of the random process $Z[n]$ equals

$$\rho_{ZZ}[m] = \mathfrak{p}_\rho(\rho_{YY}[m]) = \sum_{k=1}^{P} \xi_k \cdot \big(\rho_{YY}[m]\big)^k,$$

where $\mathfrak{p}_\rho(\cdot)$ denotes a polynomial with coefficients ξ_k, which, for $k = 1, \cdots, P$ are calculated to

$$\xi_k = \frac{1}{\sigma_Z^2} k! \left(\sum_{p=0}^{P} \alpha_p \cdot \binom{p}{k} \cdot (p-k-1)!! \cdot \mathbf{1}_e(p-k) \right)^2,$$

where σ_Z^2 denotes the variance of $Z[n]$, determined by

$$\sigma_Z^2 = \sum_{k=1}^{P} k! \left(\sum_{p=0}^{P} \binom{p}{k} \alpha_p \, (p-k-1)!! \, \mathbf{1}_e(p-k) \right)^2. \quad (15)$$

Thereby, $\binom{l}{k}$ is the binomial coefficient extended to all integer numbers (i.e., zero for $k<0$ and $k>l$), $(k)!!$ the double facto-

rial operator extend to negative values (i.e., one for $k \leq 0$) and $\mathbf{1}_e(k)$ the indicator function for parity.

B. XCF TRANSFORMATION

Let $Y_1[n]$ and $Y_2[n]$ denote two stationary Gaussian random processes with zero mean and unit variance, and $\rho_{Y_1Y_2}[m]$ the cross-correlation function between them. Both processes are transformed by polynomials into the random processes $Z_1[n]$ and $Z_2[n]$, according to $Z_1[n] = \mathfrak{p}_{Y,1}(Y_1[n]) = \sum_{p=0}^{P} \alpha_p \cdot (Y_1[n])^p$ and $Z_2[n] = \mathfrak{p}_{Y,2}(Y_2[n]) = \sum_{q=0}^{Q} \beta_q \cdot (Y_2[n])^q$, then the cross-correlation function of the processes $Z_1[n]$ and $Z_2[n]$ equals

$$\rho_{Z_1Z_2}[m] = \mathfrak{p}_{\rho,12}(\rho_{Y_1Y_2}[m]) = \sum_{k=1}^{\min(P,Q)} \chi_k \cdot \left(\rho_{Y_1Y_2}[m]\right)^k,$$

where $\mathfrak{p}_{\rho,12}(\cdot)$ denotes a polynomial with coefficients χ_k which, for $k = 1, \cdots, \min(P, Q)$ are calculated to

$$\chi_k = \frac{1}{\sigma_{Z_1}}\left(\sum_{p=0}^{P} \alpha_p \cdot \binom{p}{k} \cdot (p-k-1)!! \cdot \mathbf{1}_e(p-k)\right) \cdot \frac{1}{\sigma_{Z_2}}\left(\sum_{q=0}^{Q} \beta_q \cdot \binom{q}{k} \cdot (q-k-1)!! \cdot \mathbf{1}_e(q-k)\right) \cdot k!.$$

Here σ_{Z_1} and σ_{Z_2} denote the standard deviation of $Z_1[n]$ and $Z_2[n]$, respectively, see Eq. (15).

APPENDIX II. XCF FITTING

A. CHOLESKY-FACTORIZATION

For fitting $\mathbf{G}(B)$ by a Cholesky decomposition, an auxiliary process of random vectors $\mathbf{X}'[n]$ has to be constructed. It has $(2M+1)I$ dimensions, where M denotes the maximum lag to be modeled and I the number of output processes I. It consists of $2M+1$ shifted versions of each output process $X_i[n-m]$, with $m=-M, \ldots, M$ and $i=1, \ldots, I$, arranged according to

$$\mathbf{X}'[n] = \begin{pmatrix} (X_1[n-M], \ldots, X_1[n+M])^T \\ (X_2[n-M], \ldots, X_2[n+M])^T \\ \vdots \\ (X_I[n+M], \ldots, X_I[n+M])^T \end{pmatrix}. \quad (16)$$

The correlation matrix Γ'_X of $\mathbf{X}'[n]$ has to be constructed, whereof the single elements are all coefficients of the auto and cross-correlation functions of $X_i[n]$. Notice, that Eq. (14) must be satisfied; hence, Γ'_X is a block matrix with identity matrices on the diagonal.

Performing the Cholesky decomposition of Γ'_X yields a lower triangular matrix, which has to be normalized by $\frac{1}{\sqrt{2M+1}}$ in order to satisfy Eq. (13). Each column of this matrix can be divided into I blocks of length $2M+1$; thereby, each block is translated to one element $g_{ij}[m]$ of $\mathbf{G}[m]$, whereas each element within a block is equivalent to a coefficient of $g_{ij}[m]$ at a specific lag m with $m=-M, \ldots, M$.

Consequently, the number J of required input processes $W_j[n]$ amounts to $J=(2M+1)I$; the number of model parameters (i.e., sum of all non-zero coefficients $g_{ij}[m]$) calculates to $\frac{1}{2}\, J\, (J-1)$.

B. DIRECT FITTING

If the number of output processes is $I=2$, then it is possible to pursue a direct fitting approach. It is usually more economic than the Cholesky decomposition, both in the number of input processes $Wj[n]$ (amounting to $J=2M+2$) and the model parameters (i.e., $2J$).

It is based on the observation that Eq. (14) (i.e., zero ACF for all lags unequal to zero) can be satisfied by forcing each of the polynomials $g_{ij}(B)$, to monomials in the backshift operator B

$$g_{ij}(B) \stackrel{!}{=} g_{ij,l} \cdot B^l \quad (17)$$

where $g_{ij,l}$ denotes the only non-zero element of $g_{ij}[l]$ located at lag $m=l$. The monomial order l may vary from element to element. Thus, each element $g_{ij}[m]$ is a moving-average filter with only one timing lag. This ensures that each sample of each process $W_j[n]$ appears only once within all the samples of the process $X_i[n]$ and, consequently, does not cause any auto-correlations in $X_i[n]$.

Accordingly, the polynomial matrix $\mathbf{G}(B)$ shall be constructed as

$$c = \sum_{m=-M}^{M} |\rho_{X_1X_2}[m]| \quad (18)$$

$$g_{1,j}(B) = \text{sign}(\rho_{X_1X_2}[j-M-1]) \sqrt{c\, |\rho_{X_1X_2}[j-M-1]|} \cdot B^0$$

$$g_{2,j}(B) = \sqrt{\tfrac{1}{c}\, |\rho_{X_1X_2}[j-M-1]|} \cdot B^{j-M-1}$$

$$\mathbf{G}(B) = \begin{pmatrix} g_{1,1}(B) \cdots g_{1,2M+1}(B) & \sqrt{1-c^2}\, B^0 \\ g_{2,1}(B) \cdots g_{2,2M+1}(B) & 0 \end{pmatrix},$$

whereas sign$(\cdot)$ denotes the sign operator. In this case the targeted XCF $\rho_{X_1X_2}[m]$ is induced to the processes, without causing any auto-correlations.

C. CONSIDERING ONLY LAG ZERO

Both methods, the Cholesky-decomposition and the direct fitting, converge if the maximum lag $M=0$ and $I=2$ output processes $X_i[n]$ with $\rho_{X_1X_2}[0]=g_{21}$. Then the single parameter g_{21} is enough to specify the matrix $\mathbf{G}(B)$ by

$$\mathbf{G} = \begin{pmatrix} g_{21} & \sqrt{1-g_{21}^2} \\ 1 & 0 \end{pmatrix}, \quad \text{yielding } \Gamma = \begin{pmatrix} 1 & g_{21} \\ g_{21} & 1 \end{pmatrix}. \quad (19)$$

The restrictions on $X_i[n]$ are inherently satisfied by this fitting approach: (i) Eq. (17) is satisfied since only $g_{il}[0]$ is considered for any two processes $X_i[n]$ and $X_l[n]$. (ii) Moreover, Eq. (13) is satisfied for $\mathbf{G}$, since the cross-correlation matrix Γ has unit diagonal elements.

In order to determine the value of g_{21}, it is not required to perform the whitening procedure (see Fig. 5: Step 2C), since the correlations introduced by $h_i[m]$ equal one at lag zero. Instead, the cross-correlation coefficient $\rho_{Y_iY_l,\text{target}}[0]$ can be

directly equated with g_{21} (see Fig. 5: Step 3C),

$$g_{21} = \rho_{Y_i Y_l, \text{target}}[0]. \quad (20)$$

This yields usually better results than including the whitening procedure, since inaccuracies introduced by the model of the ACFs, are suppressed. Considering only lag zero is a parsimonious way of resembling XCFs and, thus, recommended for traffic modeling purposes.

REFERENCES

[1] R. Nelson, *Probability, Stochastic Processes, and Queueing Theory*. New York, NY, USA: Springer-Verlag, 1995.
[2] H. Che and S. Li, "Fast algorithm for measurement-based traffic modeling," *IEEE J. Sel. Areas Commun.*, vol. 16, no. 5, pp. 612–625, Jun. 1998.
[3] S. Tanwir and H. Perros, "A survey of VBR video traffic models," *IEEE Commun. Surveys Tuts.*, vol. 15, no. 4, pp. 1778–1802, Jan. 2013.
[4] Q. Zhang, "A general AR-based technique for the generation of arbitrary gamma VBR video traffic in ATM networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 9, no. 7, pp. 1130–1137, Oct. 1999.
[5] J. Faerber, "Network game traffic modeling," in *Proc. NetGames*, Braunschweig, Germany, 2002, pp. 53–57.
[6] V. Frost and B. Melamed, "Traffic modeling for telecommunication networks," *IEEE Commun. Mag.*, vol. 32, no. 3, pp. 70–81, Mar. 1994.
[7] A. Adas, "Traffic models in broadband networks," *IEEE Commun. Mag.*, vol. 35, no. 7, pp. 82–89, Jul. 1997.
[8] W. E. Leland, M. S. Taqqu, W. Willinger, and D. V. Wilson, "On the self-similar nature of Ethernet traffic (extended version)," *IEEE/ACM Trans. Netw.*, vol. 2, no. 1, pp. 1–15, Feb. 1994.
[9] M. Livny, B. Melamed, and A. Tsiolis, "The impact of autocorrelation on queuing systems," *Manag. Sci.*, vol. 39, no. 6, pp. 322–339, 1993.
[10] B. Hajek and L. He, "On variations of queue response for inputs with the same mean and autocorrelation function," *IEEE/ACM Trans. Netw.*, vol. 6, no. 5, pp. 588–598, Oct. 1998.
[11] S. Li and C. Hwang, "Queue response to input correlation functions: Discrete spectral analysis," *IEEE/ACM Trans. Netw.*, vol. 1, no. 5, pp. 678–692, Oct. 1993.
[12] A. Andersen and B. Nielsen, "On the use of second-order descriptors to predict queueing behavior of MAPs," *Naval Res. Logist.*, vol. 49, no. 4, pp. 391–409, 2002.
[13] G. Casale, N. Mi, and E. Smirni, "Bound analysis of closed queueing networks with workload burstiness," in *Proc. SIGMETRICS*, Annapolis, MD, USA, 2008, pp. 13–24.
[14] M. Laner, J. Fabini, P. Svoboda, and M. Rupp, "End-to-end delay in mobile networks: Does the traffic pattern matter?" in *Proc. ISWCS*, Ilmenau, Germany, 2013, pp. 1–5.
[15] G. Terdik, Z. Gál, and E. Iglói, "Bispectral analysis of traffic in high-speed networks," *Comput. Math. Appl.*, vol. 43, no. 12, pp. 1575–1583, 2002.
[16] P. Borgnat, P. Abry, and P. Flandrin, "Using surrogates and optimal transport for synthesis of stationary multivariate series with prescribed covariance function and non-Gaussian joint-distribution," in *Proc. IEEE ICASSP*, Kyoto, Japan, Mar. 2012, pp. 3729–3732.
[17] D. Heyman, A. Tabatabai, and T. Lakshman, "Statistical analysis and simulation study of video teleconference traffic in ATM networks," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 2, no. 1, pp. 49–58, Mar. 1992.
[18] A. Alheraish, S. Alshebeili, and T. Alamri, "A GACS modeling approach for MPEG broadcast video," *IEEE Trans. Broadcast.*, vol. 50, no. 2, pp. 132–141, Jun. 2004.
[19] A. Papoulis and S. Pillai, *Probability, Random Variables, and Stochastic Processes*, 4th ed. New York, NY, USA: McGrawn-Hill, 2002.
[20] G. Box and M. Muller, "A note on the generation of random normal deviates," *Ann. Math. Statist.*, vol. 29, no. 2, pp. 610–611, 1958.
[21] G. Box, G. Jenkins, and G. Reinsel, *Time Series Analysis: Forecasting and Control*, 4th ed. New York, NY, USA: Wiley, 2008.
[22] P. Barford and M. Crovella, "Generating representative Web workloads for network and server performance evaluation," in *Proc. SIGMETRICS*, Madison, WI, USA, 1998, pp. 151–160.
[23] W. Willinger, V. Paxson, and M. Taqqu, "Self-similarity and heavy tails: Structural modeling of network traffic," in *A Practical Guide to Heavy Tails*, R. Adler, Ed. Cambridge, MA, USA: Birkhäuser, 1998, ch. 1, pp. 27–54.
[24] P. Svoboda, "Traffic flows," in *Video and Multimedia Transmissions Over Cellular Networks: Analysis, Modelling and Optimization in Live 3G Mobile Networks*, M. Rupp, Ed. New York, NY, USA: Wiley, 2009, ch. 13.
[25] P. Svoboda, W. Karner, and M. Rupp, "Traffic analysis and modeling for world of warcraft a MMOG," in *Proc. IEEE ICC*, Glasgow, Scotland, Jun. 2007, pp. 1612–1617.
[26] I. Norros, "On the use of farctional Brownian motion in theory of connectionless networks," *IEEE J. Sel. Areas Commun.*, vol. 13, no. 6, pp. 953–962, Aug. 1995.
[27] P. Branch and G. Armitage, "Measuring the auto-correlation of server to client traffic in first person shooter games," in *Proc. ATNAC*, Melbourne, Australia, 2006, pp. 1–5.
[28] M. Dai, Y. Zhang, and D. Loguinov, "A unified traffic model for MPEG-4 and H.264 video traces," *IEEE Trans. Multimedia*, vol. 11, no. 5, pp. 1010–1023, Aug. 2009.
[29] M. Garrett and W. Willinger, "Analysis, modeling and generation of self-similar VBR video traffic," in *Proc. SIGCOMM*, London, U.K., 1994, pp. 269–280.
[30] P. Abry, P. Borgnat, F. Ricciato, A. Scherrer, and D. Veitch, "Revisiting an old friend: On the observability of the relation between long range dependence and heavy tail," *Telecommun. Syst.*, vol. 43, nos. 3–4, pp. 147–165, 2010.
[31] P. Brockwell and R. Davis, *Time Series: Theory and Methods*, 2nd ed. New York, NY, USA: Springer-Verlag, 1991.
[32] F. Bause, P. Buchholz, and J. Kriege, "A comparison of Markovian arrival and ARMA/ARTA processes for the modeling of correlated input processes," in *Proc. WSC*, Austin, TX, USA, Dec. 2009, pp. 634–645.
[33] A. Dainotti, A. Pescapé, P. S. Rossi, F. Palmieri, and G. Ventre, "Internet traffic modeling by means of hidden Markov models," *Comput. Netw.*, vol. 52, pp. 2645–2662, Oct. 2008.
[34] G. Casale, E. Z. Zhang, and E. Smirni, "Trace data characterization and fitting for Markov modeling," *Perform. Eval.*, vol. 67, no. 2, pp. 61–79, 2010.
[35] M. Menth, A. Binzenhöfer, and S. Mühleck, "Source models for speech traffic revisited," *IEEE/ACM Trans. Netw.*, vol. 17, no. 4, pp. 1042–1051, Aug. 2009.
[36] P. Branch, A. Cricenti, and G. Armitage, "A Markov model of server to client IP traffic in first person shooter games," in *Proc. IEEE ICC*, Beijing, China, May 2008, pp. 5715–5720.
[37] B. Melamed, "An overview of TES processes and modeling methodology," in *Performance Evaluation of Computer and Communication Systems*, L. Donatiello, Ed. New York, NY, USA: Springer-Verlag, 1993, ch. 1, pp. 359–393.
[38] D. L. Jagerman and B. Melamed, "The transition and autocorrelation structure of TES processes," *Commun. Statist. Stochast. Models*, vol. 8, no. 2, pp. 193–219, 1992.
[39] B. Liu and D. Munson, "Generation of a random sequence having a jointly specified marginal distribution and autocovariance," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 30, no. 6, pp. 973–983, Dec. 1982.
[40] M. Grigoriu, *Applied NonGaussian Processes*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1995.
[41] M. Cario and B. Nelson, "Numerical methods for fitting and simulating autoregressive-to-anything processes," *INFORMS J. Comput.*, vol. 10, no. 1, pp. 72–81, 1997.
[42] H. Helgason, V. Pipiras, and P. Abry, "Synthesis of multivariate stationary series with prescribed marginal distributions and covariance using circulant matrix embedding," *Signal Process.*, vol. 91, no. 8, pp. 1741–1758, 2011.
[43] W. Palma, *Long-Memory Time Series*. New York, NY, USA: Wiley, 2007.
[44] M. Laner, "Analyzing packet delay in reactive networks," Ph.D. dissertation, Inst. Telecommun., Vienna Univ. Technol., Vienna, Austria, 2013.
[45] J. Kriege and P. Buchholz, "Correlated phase-type distributed random numbers as input models for simulations," *Perform. Eval.*, vol. 68, no. 11, pp. 1247–1260, 2011.
[46] B. Biller and B. Nelson, "Modeling and generating multivariate time-series input processes using a vector autoregressive technique," *ACM Trans. Model. Comput. Simul.*, vol. 13, no. 3, pp. 211–237, 2003.

[47] X. Huang, Y. Zhou, and R. Zhang, "A multiscale model for MPEG-4 varied bit rate video traffic," *IEEE Trans. Broadcast.*, vol. 50, no. 3, pp. 323–334, Sep. 2004.

[48] C. Liew, C. Kodikara, and A. Kondoz, "MPEG-encoded variable bit rate video traffic modelling," *IEE Proc., Commun.*, vol. 152, no. 5, pp. 749–756, Oct. 2005.

[49] C. Lawson and R. Hanson, *Solving Least Squares Problems*. Englewood Cliffs, NJ, USA: Prentice-Hall, 1974.

[50] M. Matsumoto and T. Nishimura, "Mersenne twister: A 623-dimensionally equidistributed uniform pseudo-random number generator," *ACM Trans. Model. Comput. Simul.*, vol. 8, no. 1, pp. 3–30, 1998.

[51] (2014). *Parsimonious Network Traffic Modeling By Transformed ARMA Models, MATLAB Scripts* Vienna University of Technology, Institute of Telecommunication. Wien, Austria [Online]. Available: http://www.nt.tuwien.ac.at/about-us/staff/markus-laner/

[52] M. Laner, P. Svoboda, and M. Rupp, "Parsimonious fitting of long-range dependent network traffic using ARMA models," *IEEE Commun. Lett.*, vol. 17, no. 12, pp. 2368–2371, Dec. 2013.

[53] V. Paxson and S. Floyd, "Wide-area traffic: The failure of Poisson modeling," *IEEE/ACM Trans. Netw.*, vol. 3, no. 3, pp. 226–244, Jun. 1995.

[54] J. Liu, Y. Shu, L. Zhang, F. Xue, and O. W. W. Yang, "Traffic modeling based on FARIMA models," in *Proc. IEEE CCECE*, Edmonton, AB, Canada, May 1999, pp. 162–167.

[55] R. Riedi, M. S. Crouse, v. J. Ribeiro, and R. G. Baraniuk, "A multifractal wavelet model with application to network traffic," *IEEE Trans. Inf. Theory*, vol. 45, no. 3, pp. 992–1018, Apr. 1999.

[56] G. Samorodnitsky, "Long range dependence," in *Encyclopedia of Actuarial Science*, vol. 1. New York, NY, USA: Wiley, 2006, pp. 163–257.

[57] (2014). *The ACM/SIGCOMM Internet Traffic Archive* [Online]. Available: http://ita.ee.lbl.gov

[58] (2014). *OpenArena, A FPS Online Game* [Online]. Available: http://openarena.ws

[59] P. Seeling and M. Reisslein, "Video transport evaluation with H.264 video traces," *IEEE Commun. Surveys Tuts.*, vol. 14, no. 4, pp. 1142–1165, Oct. 2012.

MARKUS LANER (S'10) received the Dipl.-Ing. and Dr. techn. degrees from the Vienna University of Technology, Austria, in 2009 and 2013, respectively. Since 2007, he has been a Research Assistant with the Institute of Telecommunications, Vienna University of Technology. His research interests include data traffic modeling for communication networks and latency measurements and analysis in wireless networks.

PHILIPP SVOBODA (S'06–M'10) received the master's degree in electrical engineering and the Ph.D. degree in traffic generation in mobile cellular networks from the Vienna University of Technology, Austria, in 2004 and 2008, respectively. Currently, he holds a post-doctoral position at the Institute of Telecommunications (TC), Vienna University of Technology. His main expertise is data analysis of large data-sets and reporting in 3G/4G cellular wireless networks. He is currently teaching several courses at the TC and leading a traffic measurement and modeling work-package in the EU FP7 project Lola.

MARKUS RUPP (M'03–SM'06) received the Dipl.-Ing. degree from the University of Saarbruecken, Germany, in 1988, and the Dr.-Ing. degree from the Technical University of Darmstadt, Germany, in 1993. From 1993 to 1995, he had a post-doctoral position at the University of Santa Barbara, Santa Barbara, CA, USA. From 1995 to 2001, he was a member of technical staff with the Wireless Technology Research Department, Bell-Laboratories, Crawford Hill, NJ, USA. Since 2001, he has been a Full Professor of digital signal processing in mobile communications with the Vienna University of Technology where he founded the Christian-Doppler Laboratory for Design Methodology of Signal Processing Algorithms, Institute of Telecommunications, in 2002. He was an Associate Editor of the IEEE Transactions on Signal Processing from 2002 to 2005, and is currently an Associate Editor of the *JASP EURASIP Journal of Advances in Signal Processing* and the *JES EURASIP Journal on Embedded Systems*. He has authored or co-authored more than 450 scientific papers and patents on adaptive filtering, wireless communications, and rapid prototyping, as well as automatic design methods.