Adaptive Generative Models for Digital Wireless Channels

Salih, O.S. , Wang, Cheng-Xiang , Ai, Bo and Mesleh, Raed

Author post-print (accepted) deposited in CURVE February 2016

Original citation & hyperlink:
Salih, O.S. , Wang, Cheng-Xiang , Ai, Bo and Mesleh, Raed (2014) Adaptive Generative Models for Digital Wireless Channels. IEEE Transactions on Wireless Communications, volume 13 (9): 5173-5182
http://dx.doi.org/10.1109/TWC.2014.2325028

ISSN 1536-1276
DOI 10.1109/TWC.2014.2325028

Publisher: IEEE

Adaptive Generative Models for Digital Wireless Channels

Omar S. Salih, *Member, IEEE,* Cheng-Xiang Wang, *Senior Member, IEEE,* Yejun He, *Senior Member, IEEE,* and Bo Ai, *Senior Member, IEEE,*

Abstract

Error models that can characterize the statistical behavior of bursty error sequences in digital wireless channels are important for evaluating and designing error control strategies as well as high layer wireless protocols. Generative models have an immense impact on wireless communications industry as they can significantly reduce the computational time of simulating wireless communication links. By using a few reference error sequences obtained from a reference transmission system, adaptive generative models aim to generate many more error sequences, corresponding to various conditions of physical channels. Compared with traditional general models, this adaptive technique can further considerably reduce the computational load of generating new error sequences as there is no need to simulate the whole transmission system again. In this paper, reference error sequences are obtained by computer simulations of a long term evolution (LTE) system. Adaptive generative models are developed from several widely used generative models, namely, the simplified Fritchman model (SFM), the Baum-Welch based hidden Markov model (BWHMM), and the deterministic process based generative model (DPBGM). We produce new error sequences according to the developed adaptive generative models and compare their burst error statistics for specific channel conditions with those obtained from reference error sequences. It is demonstrated that the well-known burst error statistics of the new error sequences derived from adaptive generative models can closely match those of reference error sequences.

Index Terms

Adaptive generative models, error models, burst error statistics, digital wireless channels, Markov models.

I. Introduction

A digital (time-discrete) channel generally represents the whole wireless transmission communication chain including the transmitter, analog (or physical) channel, and receiver in the complex baseband. The input and output of a digital channel are in the digitized form. Because of impairments in wireless channels, errors frequently emerge in digital channels. Moreover, signal processing in many stages of the wireless transmission system may add further errors [1]. It is perceived that these errors arising from digital wireless channels with memory are not independent but appear in clusters or bursts. Bursty error traces can be statistically investigated and represented by mathematical channel models called error models [2]. These error models can be classified as descriptive [3] and generative [4] models. Descriptive models express the error statistics of reference error sequences obtained directly from experiments. Generative models are mechanisms that utilize the statistical properties of the bursty error sequences to generate error sequences having burst error statistics similar to those of reference error sequences. Generative models are very efficient as they decrease the computation burden of real or simulation systems and subsequently they significantly reduce the simulation time. The main application of error models is to assess the performance and assist in the design of error control schemes and also the design of high layer wireless communication protocols [5]–[9]. Error models can characterize erroneous bit or packet sequences [10], [11].

O. S. Salih is with the School of Computing, Electronics, and Maths, Faculty of Engineering, Environment, and Computing, Coventry University, Coventry, CV1 5FB, UK. E-mail: omar.salih@coventry.ac.uk

C. -X. Wang is with the Joint Research Institute for Signal and Image Processing, School of Engineering and Physical Sciences, Heriot-Watt University, Edinburgh, EH14 4AS, UK. E-mail: {cheng-xiang.wang}@hw.ac.uk.

Y. He is with the College of Information Engineering, Shenzhen University, Shenzhen 518160, China. Email: heyejun@ieee.org

B. Ai is with the State Key Laboratory of Rail Traffic Control and Safety, Beijing Jiaotong University, Beijing 100044, China. Email: aibo@ieee.org

In the literature there are five main classes of generative models. Markov models are the first class of generative models. They consist of finite [12]–[17] or infinite states [4] in a chain. The Gilbert-Elliot model [12], [13] was the first model in this category with two states for generating errors and error-free bits. Many amendments appeared afterwards in order to enhance its performance, but these models still produce error sequences with burst error statistics that diverge from the desirable ones. An increase in the number of states has achieved better performance [11], [14], [15]. For example, Simplified Fritchman's models (SFMs) [14] replaced the error-free state of the two-state model with a group of error-free states while keeping the only error state. SFMs had been applied to many systems with different types of physical channels [18], [19]. Bipartite models [11] are more advanced models, but their complexity is very high in order to acheive a satisfactory accuracy. The second class is hidden Markov Models (HMMs) [20]–[23], which contain hidden parameters that can be calibrated through observations. The Baum-Welch (BW) algorithm [24] was mostly used to attune the hidden parameters based on available observations. We call HMMs that use Baum-Welch (BW) algorithm, Baum-Welch based HMMs (BWHMMs). HMMs are considerably complicated due to the huge number of states required to train the hidden parameters, which greatly increases the computation speed. The third and fourth classes of generative models are based on stochastic context-free grammars (SCFGs) [25] and chaos theory [26]–[29], respectively. SCFGs consist of production rules and symbols; each symbol is assigned a probability that controls its behavior. These models are limited to error bursts with bell-shaped error density distributions. Chaotic generative models cannot describe the desired error correlation function with high accuracy [26]. The final class of generative models is the deterministic process based generative models (DPBGMs) [30], [31], which utilize the second order statistics of fading processes. The word 'deterministic' is used because all the parameters of the deterministic process are held constant during the simulation. DPBGMs have proven their superiority over other generated models, e.g., SFM [31]. However, DPBGMs do not construct new error bursts in the process of generating error sequences. Instead, they retrieve error bursts directly from the reference error sequences according to their lengths.

All the aforementioned traditional generative models [12]– [31] were developed and studied for error sequences of one digital channel with fixed parameters and channel conditions. However, recent applications need a high number of error patterns or sequences for different digital channels in order to efficiently evaluate the performance of error control schemes and high layer protocols. In other words, for appropriate testing of error control schemes and protocols, many error sequences of many digital channels need to be fed in the testing part in order to get performance results at various channel conditions. Obtaining many error sequences corresponding to different channel conditions, e.g., signal-to-noise-ratio (SNR) values, is very time consuming. Therefore, adaptive generative models that can utilize the available error sequences in order to attain newly required error sequences for different purposes are highly desirable.

In this paper, we investigate the useful parameters of some widely known generative models i.e., SFM, BWHMM, and DPBGM, in order to adjust them for the purpose of generating new error sequences from at least two reference error sequences obtained from a long term evolution (LTE) system. Two or more generated error sequences using the reference ones can be utilized in order to generate many more error sequences corresponding to several SNRs. Therefore, there is no need to simulate again the wireless communication system.

We summarize the contributions of this paper as:

1) Adaptive generative models are developed from three well-known generative models(SFM, BWHMM, and DPBGM).
2) The adaptively uncoded generated error sequences are fed in LTE digital channels in order to check the resulting error rate.

This paper is organized as follows. Section II defines some terms related to binary error sequences and describes some important burst error statistics as performance metrics. The novel adaptive procedures for three widely used generative models, namely SFM, BWHMM, and DPBGM, are proposed in Section III. Section IV illustrates an LTE simulator which is used as a descriptive model to derive reference bit error sequences at certain values of SNR. The burst error statistics are also compared between different generative models and descriptive models in this section. Finally, conclusions are drawn in Section V.

II. Burst error statistics

An error sequence of a digital wireless channel can be obtained by comparing the digital output sequence with the input error sequence. We will consider the bit error sequence here as a sequence of “0”s and “1”s. That means if the output bit is the same as the input bit, then this bit is represented by “0” in the error sequence. However, if the received output bit is different from the input bit, then this bit is received incorrectly and is represented by “1” in the error sequence.

We can breakdown the error sequence into smaller parts in order to study its nature and calculate the burst error statistic. In relation to this approach, a number of terms are now defined. A *gap* is defined as a sequence of consecutive zeros between two ones, having a length equal to the number of zeros [18], [32]. An *error cluster* is a series of errors that occur consecutively. It has a length equal to the number of ones [14]. An *error-free burst* is defined as an all-zero sequence with a length of at least η bits, where η is a positive integer [11], [21]. Compared to a gap, an error-free burst has the minimum length of η and is not necessarily located between two errors. An *error burst* is a series of ones and zeros restricted by “1”s at the edges, and separated from neighboring error bursts by error-free bursts [11], [21]. Clearly, the minimum length of an error burst is one and the number of consecutive error-free bits within an error burst is less than η. Hence, the local error density inside an error burst is greater than $1/\eta$.

In what follows, we list widely used burst error statistics that are available in the literature for characterizing bit error sequences:

1) $G(m_g)$: the gap distribution (GD), which is defined as the cumulative distribution function (CDF) of gap lengths m_g. This statistic gives some indication of the randomness of the channel [32].
2) $P(0^{m_0}|1)$: the error-free run distribution (EFRD), which is the probability that an error bit is followed by at least m_0 error-free bits [14]. The EFRD can be calculated from the GD [32]. Clearly, $P(0^{m_0}|1)$ is a monotonically decreasing function of m_0 such that $P(0^0|1) = 1$ and $P(0^{m_0}|1) \to 0$ as $m_0 \to \infty$. This statistic is very useful to determine the minimum error-free burst length η.
3) $P(1^{m_c}|0)$: the error cluster distribution (ECD), which is the probability that a correct bit is followed by m_c or more successive bits in error [14]. This statistic distinguishes between the bursty channels and random channels as well, i.e., bursty channels have long error clusters (e.g., 15-20), whereas random channels have short error clusters (e.g., 1-3).
4) $P_{EB}(m_e)$: the error burst distribution (EBD), which is the CDF of error burst lengths m_e. This statistic helps in designing the error bursts correcting codes [32].
5) $P_{EFB}(m_{\bar{e}})$: the error-free burst distribution (EFBD), which is the CDF of error-free burst lengths $m_{\bar{e}}$. This statistic, together with the error burst distribution, provides the basis for determining the optimum degree of interleaving with respect to a specific code [32].
6) $P(m,n)$: the block error probability distribution (BEPD), which is the probability that at least m out of n bit are in error. This statistic is important for determining the performance of Hybrid Automatic Repeat Request (HARQ) protocols [18].
7) $\rho(\Delta k)$: the bit error correlation function (BECF), which is the conditional probability that the Δkth bit following a bit in error is also in error. The BECF is also important because it represents the burstiness of the channel [3], [4].

Burst Error statistics are useful statistical means to demonstrate the natural structural behavior of error sequences obtained from wireless channels with memory. Consequently, they could help in the design and evaluation of error control schemes and higher layer protocols, especially those very important burst error statistics, namely $P(m,n)$ and $\rho(\Delta k)$. Furthermore, burst error statistics are metrics to judge the relative merits of different generative models by comparing them with the descriptive model. We will use some of these statistics in Section III to develop the new adaptive generative models and we will illustrate all of them in Section IV in order to validate our proposed generative models.

III. Adaptive generative models

Adaptive generative models are very convenient for evaluating error control schemes and high layer protocols as they can generate many new error sequences from at least two reference error sequences. This ability has a huge impact in reducing the simulation time of the original system as well as the simulation time for evaluating error control schemes. In the following subsections, we propose methods for producing new error sequences from two reference error sequences. The adopted generative models, namely the SFM, HMM, and DPBGM, are widely known in the literature and have been applied to many wireless systems.

A. *Adaptive SFM (ASFM)*

A SFM consists of N-states, one error state and $N-1$ error-free states. This division is designated in relation to very important statistics for performance evaluation, which are the error cluster distribution and error free run distribution.

When a SFM is transiting into the error state, it generates "1" (error bit). When a transition to an error-free state occurs, the SFM generates "0" (correct bit). While the SFM is circulating within an error-free state, "0"s are generated until a transition to the error state occurs. In this case, the SFM generates "1"s again. Transitions between the error-free states in a SFM are forbidden. The reason for having many states generating "0"s is to generate different lengths of gaps. All the transitions take place according to assigned probabilities. The probability transition matrix for an N-state SFM is [14]

$$\mathbf{T} = \begin{pmatrix} P_{11} & 0 & 0 & 0 & P_{1N} \\ 0 & P_{22} & 0 & 0 & P_{2N} \\ 0 & 0 & \ddots & 0 & \vdots \\ 0 & 0 & 0 & P_{N-1N-1} & P_{N-1N} \\ P_{N1} & P_{N2} & \cdots & P_{NN-1} & P_{NN} \end{pmatrix} \tag{1}$$

where P_{ij} is the the probability of transiting from State i to State j ($i, j = 1, ..., N$). Note that states $1, ..., N-1$ are error-free states, while N is the error state. As the transitions between error-free states are not allowed $P_{ij} = 0$ for $i, j = 1, ..., N-1$ and $i \neq j$. The probabilities P_{ij} can be determined from the EFRD of the reference error sequence, which is written as [14]

$$P(0^{m_0}|1) = \sum_{i=1}^{N-1} \frac{P_{Ni}}{P_{ii}} P_{ii}^{m_0}, \qquad m_0 > 0. \tag{2}$$

The EFRD can also be approximated by the weighted sum of $N-1$ exponentials given by [14]

$$P(0^{m_0}|1) \approx A_1 e^{a_1 m_0} + \cdots + A_{N-1} e^{a_{N-1} m_0}. \tag{3}$$

The parameters A_M and a_M ($M = 1, 2, \cdots, N-1$) can be found by using an optimization method or curve fitting technique to match (3) with the EFRD obtained from the reference error sequence. Consequently, the values of P_{ij} in (1) are obtained by the following [14]

$$P_{MM} = e^{a_M}, \tag{4}$$

$$P_{NM} = A_M \times P_{MM}, \tag{5}$$

$$P_{MN} = 1 - P_{MM}, \tag{6}$$

$$P_{NN} = 1 - \sum_{M=1}^{N-1} P_{NM}. \tag{7}$$

In order to generate a new error sequence, which we called it the adaptive error sequence, from reference error sequences, we have to consider the most important burst error statistic in SFM, which is the EFRD.

Once we know the new EFRD from the surrounding reference EFRDs, we can then follow the normal procedure of generating error sequences.

The procedure is simply to firstly obtain two EFRDs corresponding to two different SNRs from two reference error sequences. Subsequently, from the obtained EFRDs, we produce many new EFRDs suitable for generating many error sequences corresponding to various SNRs. Suppose we have two reference error sequences with two different SNRs in dB, e.g., X and Y, then their EFRDs at the two levels of SNR are ($P_X(0^{m_0}|1)$ and $P_Y(0^{m_0}|1)$). In order to find the new $P_Z(0^{m_0}|1)$, which is the EFRD of the new and required error sequence, we apply (see Fig. 1)

$$P_Z(0^{m_0}|1) = \lfloor P_X^{\alpha}(0^{m_0}|1) \times P_Y^{\beta}(0^{m_0}|1) \rfloor \tag{8}$$

where P_X and P_Y are weighted by

$$\alpha = \left| \frac{SNR_Z - SNR_Y}{SNR_X - SNR_Y} \right| \tag{9}$$

and

$$\beta = \left| \frac{SNR_Z - SNR_X}{SNR_X - SNR_Y} \right|, \tag{10}$$

respectively, Here, $\lfloor P \rfloor$ is the floor function of P. After obtaining $P_Z(0^{m_0}|1)$, we can simply fit it with (3) in order to find the optimized parameters A_M and a_M and consequently the transition matrix $\mathbf{T}$. Finally, the required new error sequence is ready for generation.

B. Adaptive Baum-Welch based HMM (ABWHMM)

HMMs [21], [23] employ the idea of Markov models, but use two stochastic processes. One stochastic process is not observable but can only be estimated by the other stochastic process which produces a sequence of observations. The authors of [21] implement HMMs using Baum-Welch (BW) algorithm [20], [24]. The procedure of [21] is explained as follows. The error bursts of the reference error sequence are extracted and numbered. Each error burst is then divided into blocks of L bits length. Each block is represented by the number of error bits it contains. For example, when $L = 4$, the error burst 110011110001 has 3 blocks. Hence, that error burst is represented by 3 digits as 241. In this way, the error bursts are converted into a compact format and they then form a matrix, $\mathbf{NEL}$, such that $\mathbf{NEL} = \{NEL_1, NEL_2, \cdots, NEL_m\}^{'}$, where m is the number of error bursts in the reference error sequence. The largest number in $NEL_i (i = 1, ..., m)$ is called the peak number of errors (PNE), e.g., 4 in the previous example. The next step is to classify the error bursts into N disjoint classes (submodels or bursty states) according to $\zeta(N-1)+1 \leq PNE \leq \zeta N$, where ζ is a positive integer number. Afterwards, the compacted blocks of each state shall be used to train hidden Markov submodels using BW algorithm [24]. Each submodel contains one class of error bursts. BWHMMs have the following parameters:

1) $\mathbf{S} = \{s_1, s_2, ..., s_N\}$: the set of states of the model, where N is the number of states.
2) $\mathbf{V} = \{v_1, v_2, ..., v_D\}$: the set of observable values, where D is the cardinality of the observable values.
3) $\mathbf{A} = [a_{ij}]$: the state transition probabilities matrix, where a_{ij} is the probability of transition from state s_i to s_j.
4) $\mathbf{B} = [b_{jk}]$: the observations probabilities matrix, where b_{jk} is the probability of emitting v_k from state s_j.
5) $\mathbf{\Pi} = [\pi_i]$: the initial state probability.

To build the BWHMM submodels, the parameters N, D, and the set $\lambda = \{\mathbf{A}, \mathbf{B}, \mathbf{\Pi}\}$ must be specified. The value of N can be decided according to the guidelines in [21]. Given a set of observation sequences representing the compacted error burst $\mathbf{O}^k = \left\{O_1^k, O_2^k, \cdots, O_{D_k}^k\right\}$, $k = 1, \ \cdots, \ K$ (K is the number of error bursts in each class), the BW algorithm is utilized to maximize the probability $\Gamma = \prod_{k=1}^{K} P(\lambda|\mathbf{O}^k)$.

Once the optimized transition probabilities are found out, error bursts can be generated from the submodels. To complete the generation of new error sequences, the error-free bursts concatenation to the hidden Markov submodels should be executed. The error-free bursts are represented by one state only. The transitions from the error-free state to the other states generate error bursts with variable structures according to the submodel. However, the transitions from the burst states to the error-free state generate error-free bursts with different lengths. Both error-free bursts and error bursts are combined at the end.

In order to generate many new error sequences from two reference error sequences, we should find out the most important feature of the BWHMM. It is the $\mathbf{NEL}$ matrix. In fact, error models aim to identify the error events and distribution in error bursts. This is recognized by the $\mathbf{NEL}$ matrix. From $\mathbf{NEL}$ we can know the number of errors in each block for each error burst. Therefore, from knowing two $\mathbf{NEL}$ matrices, i.e., $\mathbf{NEL}_X$ and $\mathbf{NEL}_Y$ corresponding to two different SNRs of two reference error sequences X and Y, we can obtain a new $\mathbf{NEL}$ matrix, e.g., $\mathbf{NEL_Z}$ of SNR corresponding to the new error sequence Z. The SNR of Z is between the other two SNRs of X and Y error sequences. From the new $\mathbf{NEL}$ matrix we can then generate the new required error sequence without the need for a reference error sequence for the wanted SNR. Once the $\mathbf{NEL_Z}$ matrix is calculated, the set of steps described before to construct the submodels are applicable in the process toward generating the required error sequence.

In order to find the $\mathbf{NEL_Z}$ we firstly need to sort each row in both $\mathbf{NEL_X}$ and $\mathbf{NEL_Y}$ in descending manner so that the PNE is the leading element. Secondly, the rows should be sorted so that the PNE column is in descending order as well. The $\mathbf{NEL_Z}$ can then simply be found by

$$\mathbf{NEL_Z} = \lfloor \alpha \cdot \mathbf{NEL_X} + \beta \cdot \mathbf{NEL_Y} \rfloor . \tag{11}$$

The values of α and β can be calculated from (9) and (10), respectively. Afterwards, we apply the classification rule, training procedure, and finally the concatenation method to generate the required error sequence. However, to apply the concatenation, we need to construct the error-free state. Generating new error-free bursts is discussed in the next subsection (Section IIIC).

The BWHMMs utilize the Baum-Welch algorithm because it is robust and always converges. However, the convergence point is not guaranteed to be a global maximum. Hence, its final parameters may not necessarily be the optimal ones. Another drawback is that the BWHMMs consist of a large number of states, which increases the complexity of the model.

C. Adaptive DPBGM (ADPBGM)

The idea of the DPBGM is derived from the second order statistics of fading processes [31]. Specifically, some statistics of bursty errors can be approximated from the second order statistics of fading envelope processes. Accordingly, fading processes can be used to generate error sequences. Deterministic fading processes are based on the rule of sum of sinusoids [33].

To build a DPBGM, an underlying reference transmission system is replaced by a properly parameterized and sampled deterministic process followed by a threshold detector and two parallel mappers. Mappers can fit the obtained length distributions of the error and error-free bursts to the desired statistics of the descriptive models.

The complex deterministic process can be represented by [31]

$$\tilde{\zeta}(t) = |\tilde{\mu}_1(t) + j\tilde{\mu}_2(t)| \tag{12}$$

where

$$\tilde{\mu}_i(t) = \sum_{n=1}^{N_i} c_{i,n} \cos(2\pi f_{i,n} t + \theta_{i,n}) \ , \quad i = 1, 2 \ . \tag{13}$$

Here N_i is the number of sinusoids, $c_{i,n}$ are gains, $\theta_{i,n}$ are phases for the realizations of the random generators, and $f_{i,n}$ are the discrete frequencies. Some second order statistics of the sampled deterministic process, such as the level crossing rate (LCR), the average duration of fades (ADF), the average duration of the inter-fades (AIDF), can be described using the vector $\Psi = (N_1, N_2, r_{th}, \sigma_0, f_{max}, T_A)$, where r_{th} is the threshold, $\sigma_0 = \frac{r_{th}}{\sqrt{2\ln(1+\mathcal{R}_B)}}$ is the square root of the mean power of $\mu_i(t)$, $f_{max} = \frac{\mathcal{N}_{EB}(1+\mathcal{R}_B)}{T_t\sqrt{2\pi\ln(1+\mathcal{R}_B)}}$ is the maximum Doppler frequency, and $T_A \approx \frac{4\sigma_0[\exp(\frac{r_{th}^2}{2\sigma_0^2})-1]}{\sqrt{5}\pi r_{th} f_{max}}\sqrt{-1+\sqrt{1+10q_s/3}}$ is the sampling interval. The value of $\mathcal{R}_B$ is the ratio of the mean value of error burst lengths to the mean value of error-free burst lengths. The parameters $\mathcal{N}_{EB}$ and T_t are the total number of error bursts and the total transmission time of the communications system, respectively. The quantity q_s is the maximum measurement error of the LCR.

When the simulation is run, the deterministic process varies in a way that it crosses the threshold with positive and negative slopes. When the level of the deterministic process is above that threshold (inter-fade intervals) an error-free burst is generated. On the contrary, when the deterministic level is below the threshold (fading intervals) an error burst is generated. The lengths of the error-free bursts and error bursts equal the number of samples counted in inter-fading and fading intervals, respectively. Subsequently, error burst and error-free burst generators are produced. After that, mapping [31] is employed to adjust the generated error and error-free bursts lengths to those of the original error sequence. Subsequently, we collate an error burst record $\mathbf{EB}_{rec}$ and error-free burst record $\mathbf{EFB}_{rec}$ as vectors. Finally, error sequences can be obtained by combining the consecutively generated error bursts with error-free bursts.

Let us denote the minimum value in $\mathbf{EB}_{rec}$ as m_{B1} and the maximum value as m_{B2}. Subsequently, the lengths m_e of error bursts satisfy $m_{B1} \leq m_e \leq m_{B2}$. By analogy, the minimum value and the maximum value in $\mathbf{EFB}_{rec}$ are denoted as $m_{\bar{B}1}$ and $m_{\bar{B}2}$, respectively, and the lengths $m_{\bar{e}}$ of error-free bursts satisfy $m_{\bar{B}1} \leq m_{\bar{e}} \leq m_{\bar{B}1}$. For the convenience of developing the ADPBGMs, the following quantities are defined:

1) $\mathcal{N}_{EB}$ is the total number of error bursts, which equals the number of entries in $\mathbf{EB}_{rec}$.
2) $\mathcal{N}_{EFB}$ is the total number of error-free bursts, which equals the number of entries in $\mathbf{EFB}_{rec}$.
3) $N_{EB}(m_e)$ is the number of error bursts of length m_e in $\mathbf{EB}_{rec}$. Thus, $\sum_{m_e=m_{B1}}^{m_{B2}} N_{EB}(m_e) = \mathcal{N}_{EB}$ holds.
4) $N_{EFB}(m_{\bar{e}})$ is the number of error-free bursts of length $m_{\bar{e}}$ in $\mathbf{EFB}_{rec}$. Similarly, $\sum_{m_{\bar{e}}=m_{\bar{B}1}}^{m_{\bar{B}2}} N_{EFB}(m_{\bar{e}}) = \mathcal{N}_{EFB}$ holds.

In order to design the ADPBGM, we focus on $\mathbf{EB}_{rec}$ and $\mathbf{EFB}_{rec}$ since the most important features of this model are the lengths of error bursts and error-free bursts. The first step is to calculate the two $\mathbf{EB}_{rec}$ and two $\mathbf{EFB}_{rec}$ for the two different SNRs corresponding to X and Y generated error sequences, respectively. Our aim is to calculate the $\mathbf{EB}_{rec}$ and $\mathbf{EFB}_{rec}$ that are related to the adaptive error sequence Z. Hence, the second step is to find the CDFs P_{EB} and P_{EFB} for both $\mathbf{EB}_{rec}$ and $\mathbf{EFB}_{rec}$ related to error sequences X and Y. A very important property of P_{EB} and P_{EFB} is that they are monotonically increasing. This property simplifies finding accurate values (new curves) between two P_{EB} curves (see Fig. 2) or between two P_{EFB} curves. The third step is to find P_{EB_Z}, which is related to the adaptive error sequence, from the P_{EB_X} and P_{EB_Y}, which are related to X and Y error sequences having different SNRs, by simply applying

$$P_{EB_Z} = \lfloor \alpha \cdot P_{EB_X} + \beta \cdot P_{EB_Y} \rfloor \tag{14}$$

where $P_{EB} = \frac{1}{\mathcal{N}_{EB}} \sum_{m_{B1}}^{x=m_e} N_{EB}(x)$. By analogy

$$P_{EFB_Z} = \lfloor \alpha \cdot P_{EFB_X} + \beta \cdot P_{EFB_Y} \rfloor, \tag{15}$$

where $P_{EFB}(m_{\bar{e}}) = \frac{1}{\mathcal{N}_{EFB}} \sum_{m_{\bar{B}1}}^{x=m_{\bar{e}}} N_{EFB}(x)$. The values of α and β are obtained from (9) and (10), respectively. The fourth step is to construct $\mathbf{EB}_{rec}$ and $\mathbf{EFB}_{rec}$ from P_{EB_Z} and P_{EFB_Z}, respectively. In order to do so, we have to know the total numbers of error bursts $\mathcal{N}_{EB_Z}$ and error-free bursts $\mathcal{N}_{EFB_Z}$ of the $\mathbf{EB}_{rec}$ and $\mathbf{EFB}_{rec}$ related to Z. The number $\mathcal{N}_{EB_Z}$ is obtained by interpolating between the $\mathcal{N}_{EB_X}$ and $\mathcal{N}_{EB_Y}$. Moreover, the number $\mathcal{N}_{EFB_Z}$ is obtained by interpolating between the $\mathcal{N}_{EFB_X}$ and $\mathcal{N}_{EFB_Y}$ given that X and Y have the same length. Multiplying the obtained numbers with the extracted P_{EB_Z} and P_{EFB_Z}, respectively, with some manipulations related to the CDF gives us the required $\mathbf{EB}_{rec}$ and $\mathbf{EFB}_{rec}$ with m_e and $m_{\bar{e}}$ values equivalent to those of the reference records. The fifth step is to generate error bursts and error-free bursts according to the lengths in the obtained $\mathbf{EB}_{rec}$ and $\mathbf{EFB}_{rec}$. Generating error-free bursts is simple because the lengths of $\mathbf{EFB}_{rec}$ can easily converted to series of zeros, unlike error bursts which contain zeros and ones. Generating error bursts involves retrieving their structures from the error bursts of X and Y, which have the same m_e as in the obtained $\mathbf{EB}_{rec}$ of Z. Finally, the error bursts and error-free bursts are combined to construct the required adaptive generated error sequence Z. It is worth mentioning that the DPBGM is a recent and promising class of error models. It yields on satisfactory match to the important burst error statistics compared with those of the original error sequences. Furthermore, the DPBGM parameters can easily be determined, its process can effectively be implemented using the computer, and the statistical properties can be varied over a wide range. The DPBGM has a drawback in the stage of generating error sequences, because it always needs to retrieve error bursts from reference error sequences rather than to construct them by itself. In contrast, the other methods, especially those based on Markov models construct the error bursts intuitively within the error sequence generation.

IV. Simulation Results and Discussions

To validate our proposed adaptive generative models, we first need to generate some error sequences based on their reference error sequences. These reference error sequences are essential to initialize various parameters for the generative models. We use an LTE system to obtain the required reference error sequences. The performance criteria are evaluated by calculating the burst error statistics that are defined in Section II. Our model is optimal if the obtained burst error statistics from the generative models match the descriptive model, especially the most important statistics such as the BEPD which is useful for designing and evaluating some digital components in the wireless communication chain.

The LTE system [34] consists of a turbo encoder, a burst interleaver, a rate matcher, and adaptive modulation and coding (AMC), a viterbi equalizer, a burst deinterleaver, a turbo decoder, as well as a cyclic redundancy check (CRC) for error detection at the receiver side. The utilized propagation channel can be expressed as NAMEx, where x represents the vehicle speed in km/h. NAME here represents the name of the underlying channel, e.g., a rural area (RA) channel, a typical urban (TU) channel, or a pedestrian B (PedB) channel. We use the following channels: RA275, TU3, TU50, PedB5, and PedB10. The data were transmitted as uncoded bits of length 12×10^6 with a transmission rate of $F_s = 3450$ kb/s. Target error sequences were produced at SNRs between 1 dB to 7 dB with unit step increment.

By comparing the transmitted error sequence with the received one, we workout the bit error sequences. We use the three discussed generative models, namely, the SFM, BWHMM, and DPBGM in order to generate new error sequences of length 15×10^6 bits based on the obtained error sequences from the LTE system. In this paper, we show only the results of the TU50 channel having SNRs of 2, 3, 4, and 5 dB. In order to examine the adaptivity of our procedure we produce an error sequence of 4 dB from those already generated at 3 dB and 5 dB SNRs (First scenario, Figs. 3–7). We also produce an error sequence of 4 dB from error sequences of 2 dB and 5 dB SNRs (Second scenario, Figs. 8–11). We compare the burst error statistics of the former and latter produced error sequences with those statistics obtained from the reference error sequence of the LTE simulator having SNR of 4 dB. In terms of parameterization, the value of η can be found from Fig. 1 when the curve is tending to turn. The value of η is chosen to be 20 for all our shown results.

For SFM, the fitting of $P(0^{m_0}|1)$ is achieved by using five exponentials and therefore, $N = 6$ holds. In our experiments, no better performance can be accomplished for SFMs with more than six states. After we fit Eq. (3) with $P(0^{m_0}|1)$ of SNRs of 2 dB, 3 dB, and 5 dB, we can obtain the transition matrices from which we can generate new error sequences. Afterwards, we apply Eqs. (9), (10), and (8) to calculate the adaptive $P(0^{m_0}|1)$ having SNR of 4 dB. This means that $\alpha = \beta = 0.5$ for the first scenario and $\alpha = 1/3$ and $\beta = 2/3$ for the second scenario. Once we know the adaptive $P(0^{m_0}|1)$, Eqs. (3) and (1) can be applied to generate the new error sequences.

For BWHMM, we first extract the error bursts from the error sequences of 3 dB and 5 dB SNRs. Then, we divide each error burst into blocks with $L = 20$ bits. Then, we can obtain the $\mathbf{NEL}$ matrices. We apply Eq. (11) afterwards to obtain the $\mathbf{NEL}$ matrix between the other matrices. A Baum-Welch training process will then be applied to the newly obtained $\mathbf{NEL}$ matrix after classifying it into a satisfactory number of states. The number of classes (states), in our example is 7. The number of substates is considerably large. Finally, the generated error burst will be concatenated with the generated error-free bursts in order to produce the full required error sequence. The generated error-free burst are obtained through calculating Eq. (14).

In order to proceed with the DPBGM, we need to find the vector, $\mathbf{\Psi}$, which is the set of parameters to generate the error sequences. The value of q_s is chosen to be 0.01. For the error sequences with SNRs of 2 dB, 3 dB, and 5 dB, the values of $\mathcal{N}_{EB} = 428418, 69706, 122474$ and $\mathcal{R}_B = 8.43, 5.24, 2.09$, in sequence. Consequently, $\mathbf{\Psi}_2$ = (9, 10, 0.09, 0.0425, 34.9 kHz, 8.33 μs), $\mathbf{\Psi}_3$ = (9, 10, 0.09, 0.0470, 36.8 kHz, 5.43 μs), and $\mathbf{\Psi}_5$ = (9, 10, 0.09, 0.0599, 40.9 kHz, 4.97 μs). Once we generate error sequences for the above SNRs, we can use their error burst lengths and error-free burst lengths. By calculating their P_{EB} and P_{EFB} and applying Eqs. (13) and (14), the new error burst and error-free burst lengths for SNR of 4 dB can be easily obtained. Eventually, the error bursts and error-free bursts are combined together to construct the entailed error sequences. The structure of bits in error bursts is also retrieved from the other two surrounding error sequences based on the error burst lengths.

Figs. 3–11 depict the performance of adaptive generative models. The figures also illustrate the discrepancy between different generative models, namely the SFM, BWHMM, and DPBGM. Various burst error statistics such as the EBDs, EFBDs, EFRDs, GDs, ECDs, BEPDs, and BCFs are investigated (some are shown and some are not shown due to the available space). Figs. 3-7 are related to the generated error sequence of 4 dB from the neighboring error sequences of 3 and 5 dB. Those figures omit the comparison between the DPBGM and ADPBGM since the DPBGM gives approximate results to the descriptive one [31], and also for the sake of clarity. Here, as mentioned before, we use three error sequences with different SNRs, namely, 3 dB, 4 dB, and 5 dB depicting different digital channels. We compare the burst error statistics of the error sequence obtained by the adaptive generative models with those of an error sequence having the same SNR but obtained directly from the LTE system. Generally, the ADPBGM shows the best fit to the descriptive model which represents a real reference error sequence. This is clear for all the burst error statistics except a small mismatch at the end of the curve for ECD and BEPD. The second best generative model is the ABWHMM. However, the shown burst error statistics are not comparable with ADPBGM. The ABWHMM results are slightly worse than those obtained using the normal BWHMM procedure. It is not worth comparing the ASFM with the descriptive model as the mismatch is huge. However, the ASFM and SFM comparison demonstrates a perfect in the burst error statistics.

The ADPBGM and ASFM burst error statistics match those of the DPBGM and SFM, respectively, because their main characteristics, i.e., EBD and EFRD, respectively, which are used to design our procedure, have a certain known form following a monotonically increasing or decreasing function. However, the ABWHMM main characteristic is a matrix from which it is difficult to derive a new accurate matrix using the interpolation methods. In general, the adaptive generative models are very efficient in terms of accuracy in addition to saving the simulation time as there is no need to start the generation process from the beginning each time we need new error sequences. Obtaining a reference error sequence with length of 20 million bits takes hours. However, generating an error sequence takes a few minutes, whereas using adaptive generative models takes a few seconds.

We also examine the ADPBGM by producing error sequences of 4 dB from other error sequences of 2 and 5 dB as shown in Figs. 8–11. It is found that, distancing the SNRs that are required to produce the new error sequence, deteriorates the performance. Fig . 8 also illustrates the production of 4dB BEPD by the ASFM using 2 and 5 dB error sequences. It is apparent that the ASFM is not affected by distancing the reference SNRs. This is because the required EFRD to parametrize the ASFM can be obtained by any pair of other EFRDs of different SNRs. Fig. 12 shows the coded BER curves after feeding the generated error sequences obtained from neighboring error sequences such that $\alpha = \beta = 0.5$. It is apparent that the ADPBGM outperforms the other models.

V. Conclusions

In this paper, we have proposed general methods for extracting adaptive generative error sequences without the need of their reference error sequences given that a few surrounding reference error sequences are available. Adaptive generative models are important because the designer does not need to refer to the original system in order to derive new error sequences when the channel conditions are changing. Therefore, these methods can significantly reduce the computation time when there is a need for huge number of error sequences for the purpose of evaluating the performance of digital components in communication links. At least two reference error sequences in different channel conditions should be sufficient for the method presented in this work.

To validate our proposed method, we have used uncoded LTE system to obtain a few samples of reference error sequences at various SNRs. It has been illustrated through simulations that the ADPBGM can approximately fit the descriptive model. Other adaptive generative models like the ABWHMM and ASFM give poor burst error statistics compared to the descriptive model. However, the ABWHMM is superior to the ASFM in terms of certain burst error statistics. In other words, the ABWHMM performance is closer to the descriptive model than the ASFM one. It is also found that the burst error statistics of the ASFM match those of the SFM. However, the burst error statistics of the ABWHMM do not have a satisfactory match to those of the BWHMM.

A drawback of ADPBGM is that it retrieves the error bursts' structure from the neighboring error sequences. In contrary, the ABWHMM and ASFM can create the error bursts and error-free bursts automatically once the required parameters are calculated.

ACKNOWLEDGMENTS

The authors would like to acknowledge the sponsorship of this work by the EPSRC and Philips Research Cambridge, UK, and the support from the RCUK for the UK-China Science Bridges Project: R&D on (B)4G Wireless Mobile Communications. O. S. Salih and C.-X. Wang would also like to acknowledge the support from the Scottish Funding Council for the Joint Research Institute in Signal and Image Processing with the University of Edinburgh, as a part of the Edinburgh Research Partnership in Engineering and Mathematics (ERPem).

REFERENCES

[1] J. G. Proakis, Ed., *Digital Communications*. 4th edition, New York: Mc Graw Hill, 2001.
[2] H.-H. Hung and L.-J. Chen, "An analytical study of wireless error models for bluetooth networks," in *Proc. IEEE AINA'08*, GinoWan, Japan, Mar. 2008, pp. 1317–1322.
[3] P. M. Crespo, R. M. Pelz, and J. Cosmas, "Channel error profiles for DECT," *IEE Proceedings-Commun.*, vol. 141, no. 6, pp. 413–420, Dec. 1994.
[4] L. N. Kanal and A. R. K. Sastry, "Models for channels with memory and their applications to error control," *Proc. of the IEEE'77*, vol. 66, no. 7, pp. 724–744, July 1977.
[5] M. Zorzi, R. R. Rao, and L. B. Milstein, "ARQ error control for fading mobile radio channels," *IEEE Trans. Veh. Technol.*, vol. 46, no. 2, pp. 445–455, May 1997.
[6] M. Zorzi and R. R. Rao, "Perspectives on the impact of error statistics on protocols for wireless networks," *IEEE Personal Commun.*, vol. 6, no. 10, pp. 32–40, Oct. 1999.
[7] G. Mazzini, "Queue system with smart ARQ on Markov channel for energy efficiency." in *Proc. IWCMC'05*, Jun. 2005, pp. 546–551.
[8] L. Badia, N. Baldo, M. Levorato, and M. Zorzi, "A Markov framework for error control techniques based on selective retransmission in video transmission over wireless channels," *IEEE J. Sel. Areas Commun.*, vol. 28 , no. 3, pp. 488–500, Apr. 2010.
[9] L. Badia, "On the effect of feedback errors in Markov models for SR ARQ packet delays," in *Proc. IEEE Globecom'09*, Hawaii, USA, Dec. 2009, pp. 3745–3750.
[10] C. Jiao, L. Schwiebert, and B. Xu, "On modeling the packet error statistics in bursty channels," in *Proc. LCN'02*, Tamba, Florida, USA, Nov. 2002.
[11] A. Willig, "A new class of packet- and bit-level models for wireless channels," *Proc. IEEE PIMRC'02*, Lisbon, Portugal, Sept. 2002, pp. 2434–2440.
[12] E. N. Gilbert, "Capacity of a burst-noise channel," *Bell Syst. Tech. J.*, vol. 39, pp. 1253–1265, Sept. 1960.
[13] E. O. Elliot, "Estimates of error rates for codes on burst-noise channels," *Bell Syst. Tech. J.*, vol. 42, pp. 1977–1997, Sept. 1963.
[14] B. D. Fritchman, "A binary channel characterization using partitioned Markov chains," *IEEE Trans. Information Theory*, vol. 13, no. 2, pp. 221–227, Apr. 1967.
[15] H. S. Wang and N. Moayeri, "Finite-state Markov channel - a useful model for radio communication channels," *IEEE Trans. on Veh. Technol.*, vol. 44, pp. 163-171, Feb. 1995.
[16] F. De Rango, F. Veltri, S. Marano, "Channel modeling approach based on the concept of degradation Level discrete-time Markov chain: UWB system case study," *IEEE Trans. Wireless Commun.*, vol. 10, no. 4, pp. 1098–1107, Apr. 2011.
[17] K. Fukawa, H. Suzuki, and Y. Tateishi "Packet error rate analysis using Markov models of signal-to-interference ratio for mobile packet systems," *IEEE Trans. Veh. Techol.*, vol. 61, no. 6, pp. 2517–2530, Jul. 2012.
[18] F. Swarts and H. C. Ferreira, "Markov characterization of digital fading mobile VHF channels," *IEEE Trans. Veh. Technol.*, vol. 43, no. 4, pp. 977–985, Nov. 1994.
[19] A. Semmar, M. Lecours, J. Y. Chouinard, and J. Ahern, "Characterization of error sequences in UHF digital mobile radio channels," *IEEE Trans. Veh. Technol.*, vol. 40, no. 4, pp. 769–776, Nov. 1991.
[20] W. Turin, *Digital Transmission Systems: Performance Analysis and Modeling*. New York: McGraw-Hill, 1999.
[21] J. Garcia-Frias and P. M. Crespo, "Hidden Markov models for burst error characterization in indoor radio channels," *IEEE Trans. Veh. Technol.*, vol. 46, no. 6, pp. 1006–1020, Nov. 1997.
[22] O. S. Salih, C.-X. Wang, D. I. Laurenson,"Double embedded processes based hidden Markov models for binary digital wireless channels," in *Proc. ISWCS'08*, Reykjavik, Iceland, Oct. 2008, pp. 219–223.
[23] M. U. Ilyas and H. Radha, "Modeling, estimating and predicting the packet-level Bit Error Rate process in IEEE 802.15.4 LR-WPANs using Hidden Markov Models, " in *Proc. IEEE CISS'09*, Biltamore, MD, USA, Mar. 18–20, 2009, pp. 241–246.
[24] L. R. Rabiner, "A tutorial on hidden Markov models and selected applications in speech recognition," *in Proc. IEEE'89*, vol. 77, pp. 257–286, Feb. 1989.
[25] W. Zhu and J. Garcia-Frias, "Stochastic context-free grammars and hidden Markov models for modeling of bursty channels," *IEEE Trans. Veh. Technol.*, vol. 53, no. 3, pp. 666–676, May 2004.
[26] E. Costamagna, L. Favalli, and P. Gamba, "Multipath channel modeling with chaotic attractors," *Proc. of the IEEE'02*, vol. 90, no. 5, pp. 842–859, May 2002.
[27] E. Costamagna, L. Favalli, P. Gamba, and P. Savazzi, "Block-error probabilities for mobile radio channels derived from chaos equations," *IEEE Commun. Letters*, vol. 3, no. 3, pp. 66–68, March 1999.

[28] E. Costamagna, L. Favalli, P. Savazzi, and F. Tarantola, “Long sequences of error gaps derived from chaotic generators optimized for short ones in mobile radio channels,” *Proc. IEEE VTC’04-Fall*, Los Angeles, USA, Sept. 2004.
[29] A. Köpke, A. Willig, and H. Karl, “Chaotic maps as parsimonious bit error models of wireless channels,” in *Proc. IEEE INFOCOM’03*, San Francisco, USA, Mar. 30–Apr. 3, 2003, pp. 513–523.
[30] C. X. Wang and M. Pätzold, “A generative deterministic model for digital mobile fading channels,” *IEEE Commun. Letters*, vol. 8, no. 4, pp. 223–225, Apr. 2004.
[31] C.-X. Wang and W. Xu, “A new class of generative models for burst-error characterization in digital wireless channels,” *IEEE Trans. Commun.*, vol. 55, no. 3, pp. 453–462, Mar. 2007.
[32] S. Tsai, “Markov characterization of the HF channel,” *IEEE Trans. Commun. Technol.*, vol. 17, no. 1, pp. 24–32, Feb. 1969.
[33] W. C. Jakes, Ed., *Microwave Mobile Communications*. 2nd edition, New Jersey: IEEE Press, 1994.
[34] C. Mehlfhrer, M. Wrulich, J. C. Ikuno, D. Bosanska, and M. Rupp, “Simulating the long term evolution physical layer, *Proc. EUSIPCO’09*, Glasgow, UK, Aug. 2009, pp. 1471–1478.

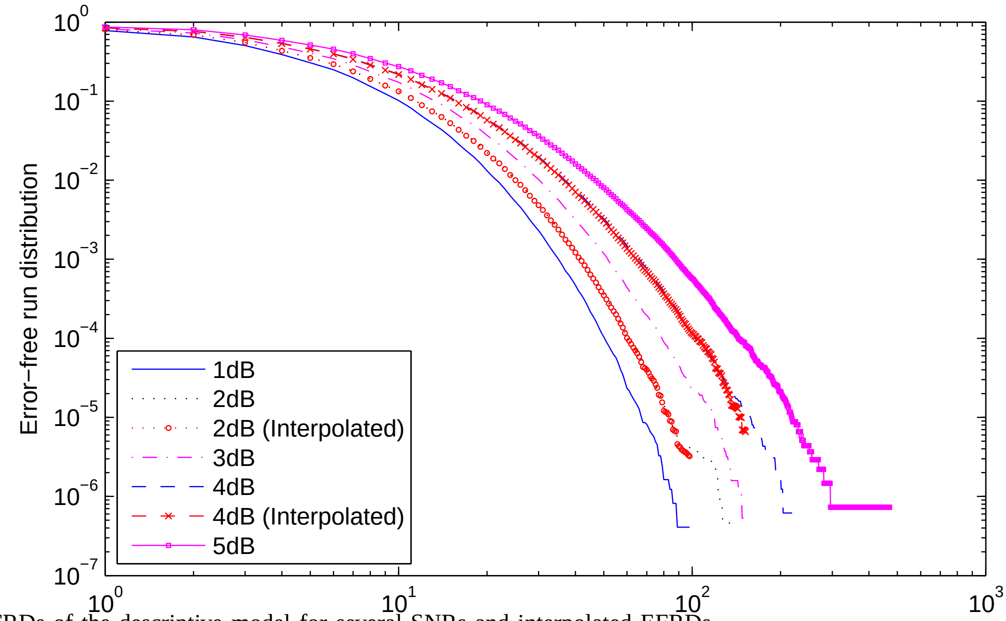

Fig. 1. The EFRDs of the descriptive model for several SNRs and interpolated EFRDs.

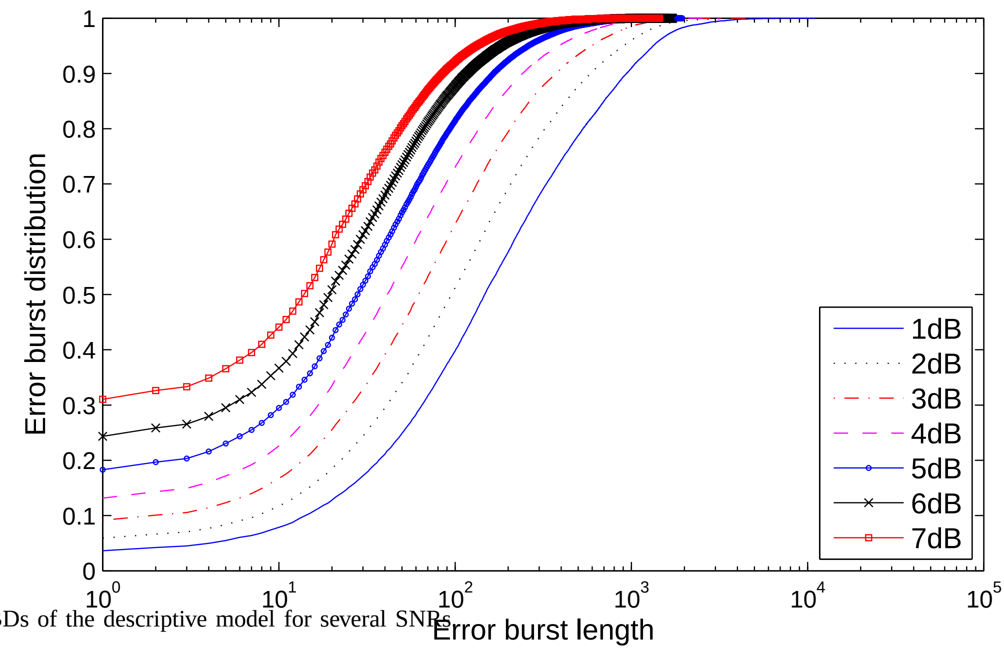

Fig. 2. The EBDs of the descriptive model for several SNRs.

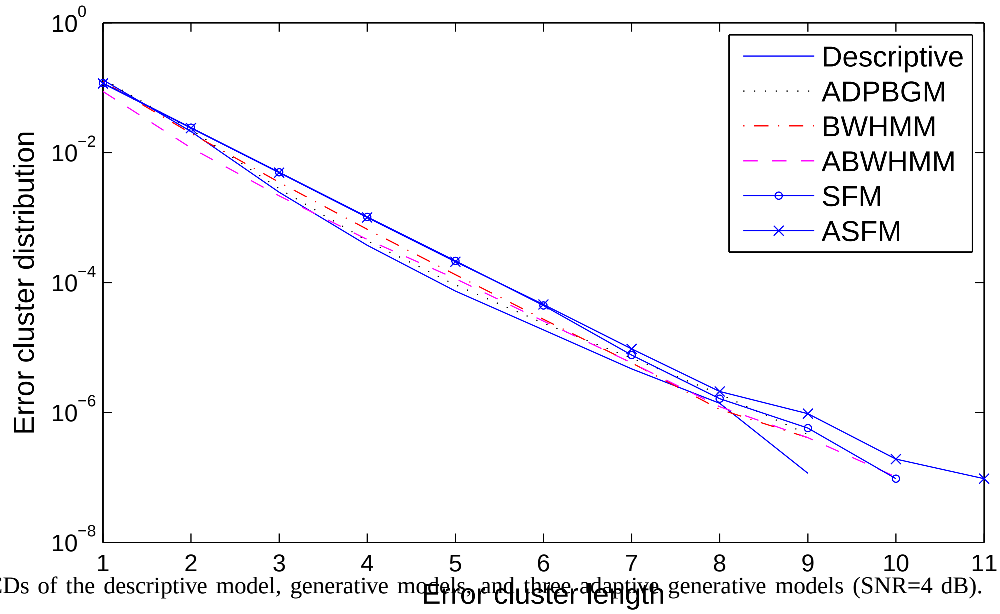

Fig. 3. The ECDs of the descriptive model, generative models, and three adaptive generative models (SNR=4 dB).

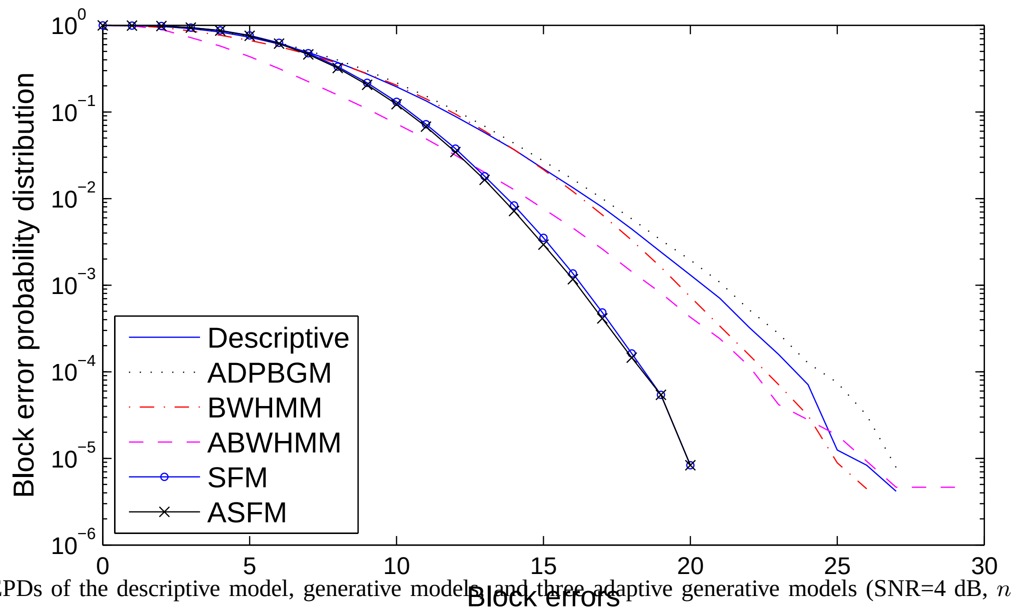

Fig. 4. The BEPDs of the descriptive model, generative models, and three adaptive generative models (SNR=4 dB, $n = 50$).

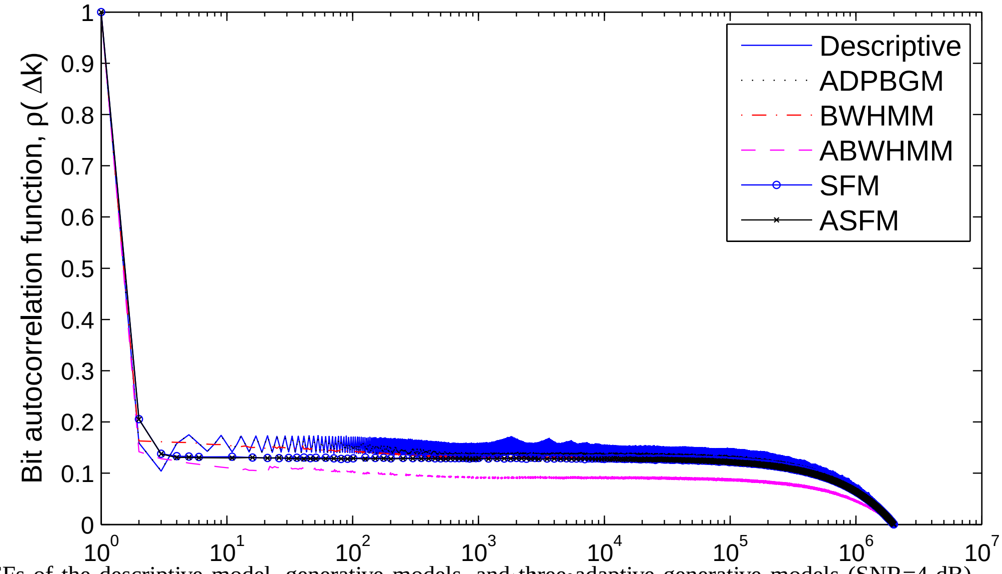

Fig. 5. The BCFs of the descriptive model, generative models, and three adaptive generative models (SNR=4 dB).

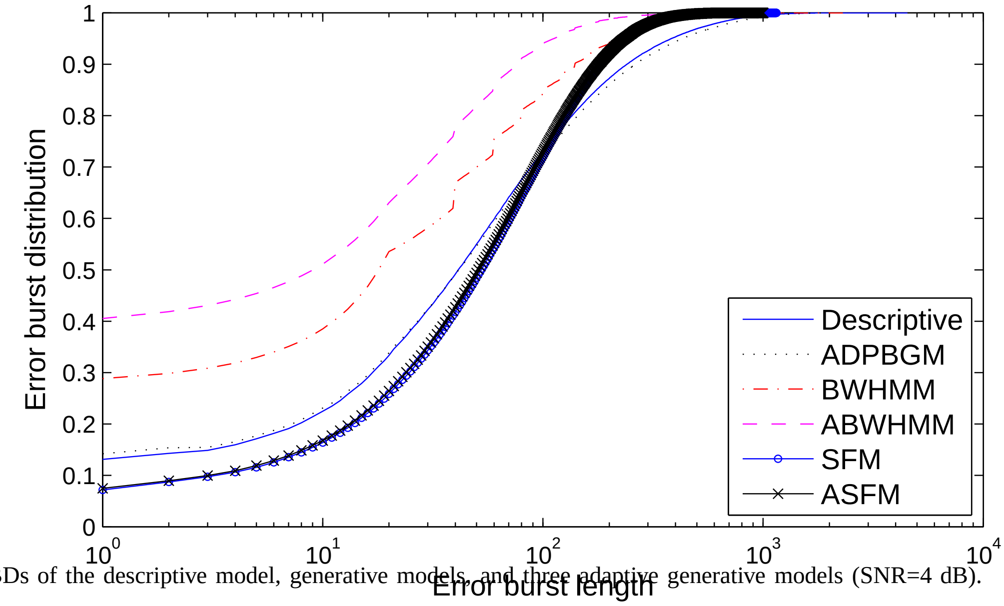

Fig. 6. The EBDs of the descriptive model, generative models, and three adaptive generative models (SNR=4 dB).

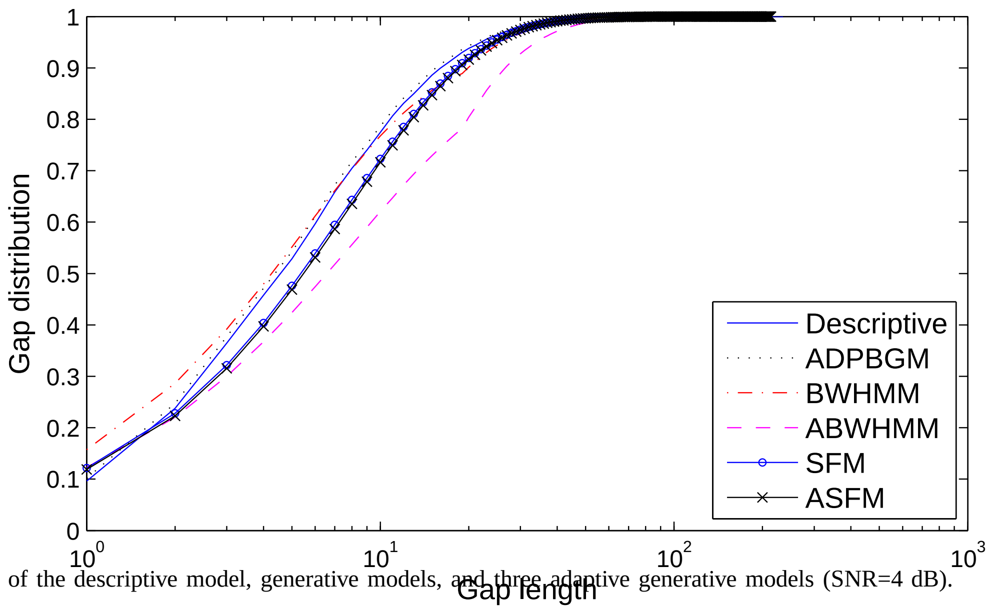

Fig. 7. The GDs of the descriptive model, generative models, and three adaptive generative models (SNR=4 dB).

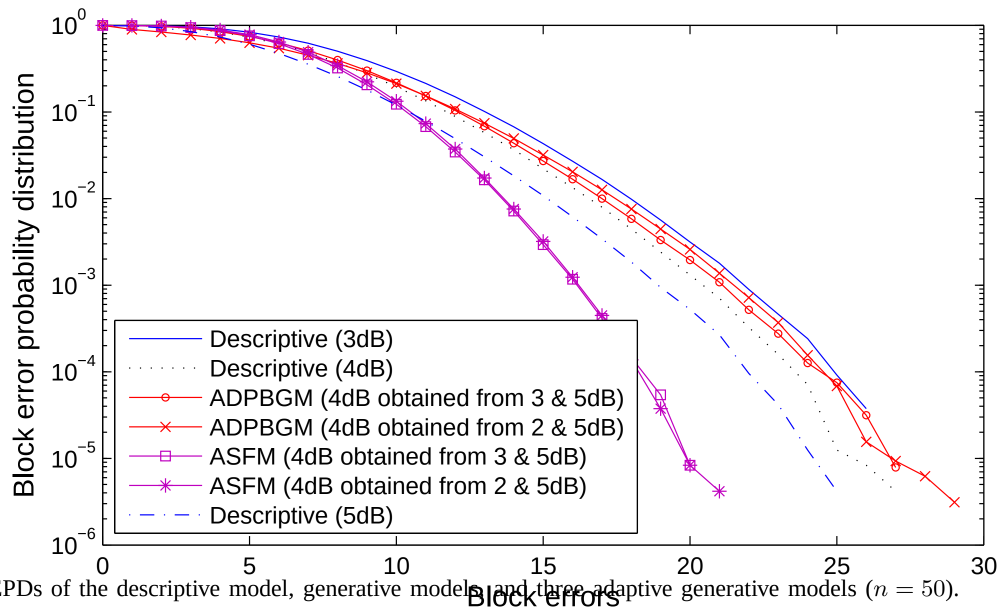

Fig. 8. The BEPDs of the descriptive model, generative models, and three adaptive generative models ($n = 50$).

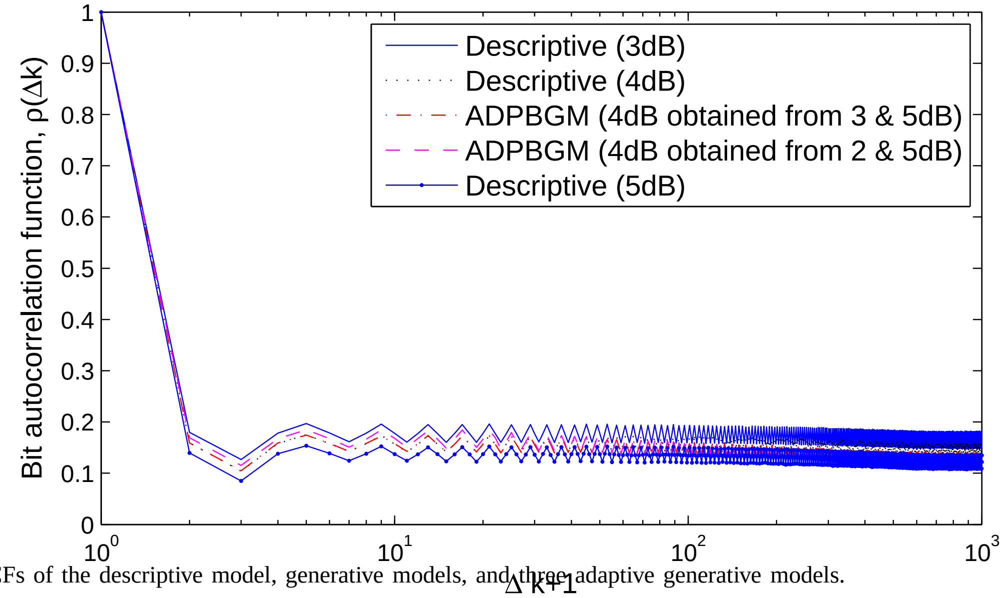

Fig. 9. The BCFs of the descriptive model, generative models, and three adaptive generative models.

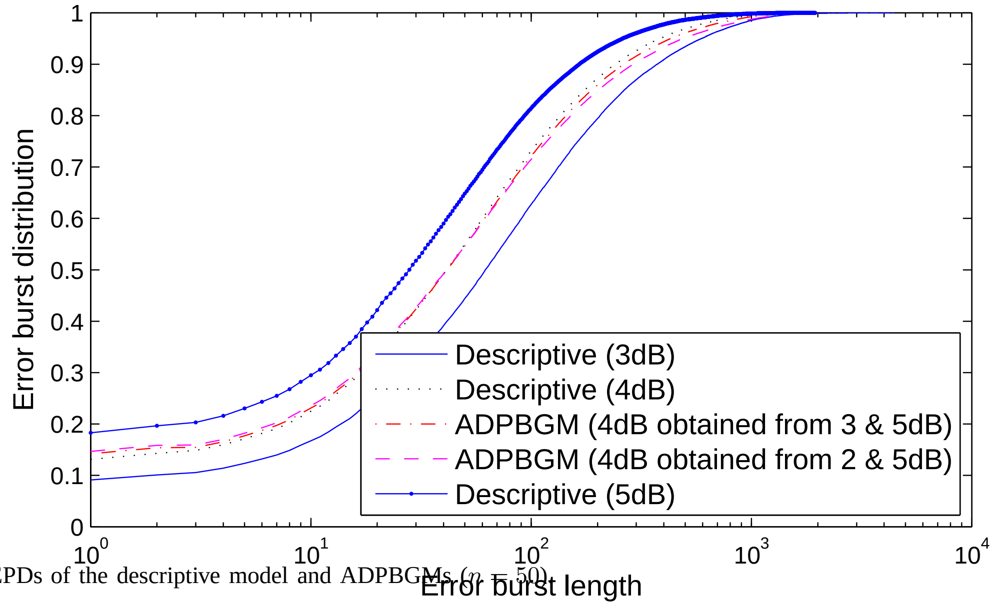

Fig. 10. The BEPDs of the descriptive model and ADPBGMs ($n = 50$).

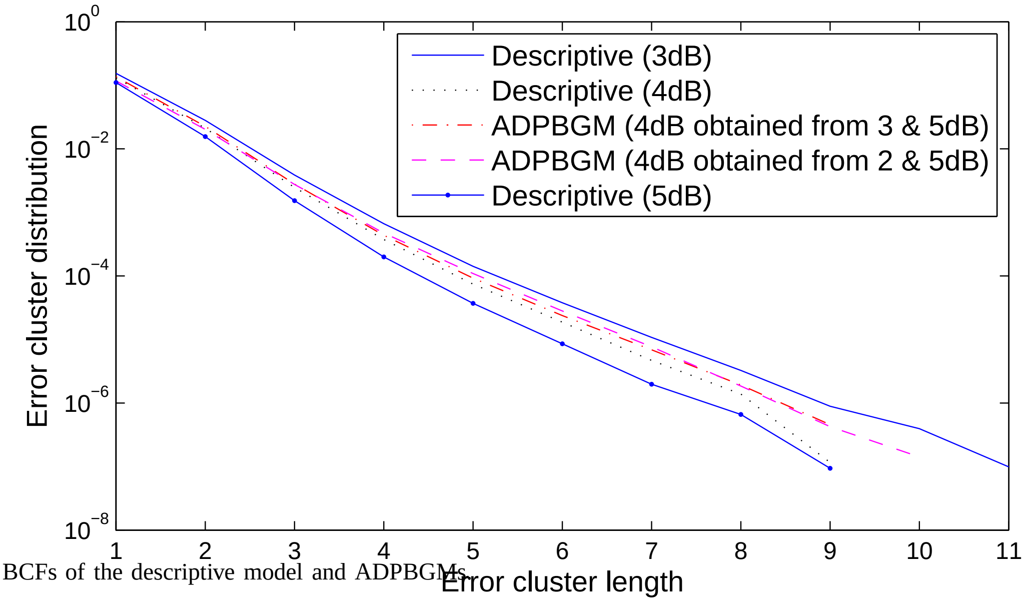

Fig. 11. The BCFs of the descriptive model and ADPBGMs.

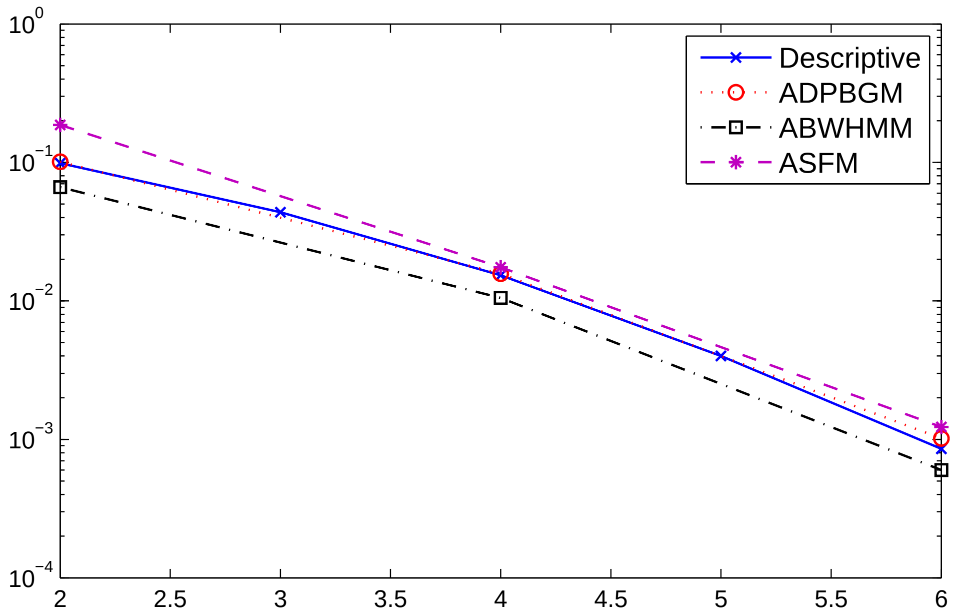

Fig. 12. The coded BER of the descriptive model and three adaptive generative models.