Hidden Markov Models for Packet-Level Errors in Bursty Digital Wireless Channels

Omar S. Salih #1, Cheng-Xiang Wang #2, David I. Laurenson *3, Yejun He **4

Joint Research Institute for Signal and Image Processing, Heriot-Watt University, Edinburgh, EH14 4AS, UK
[1] uss3@hw.ac.uk, [2] cheng-xiang.wang@hw.ac.uk

* *Joint Research Institute for Signal and Image Processing, University of Edinburgh, Edinburgh, EH9 3JL, UK*
[3] dave.laurenson@ed.ac.uk

* *College of Information Engineering, Shenzhen University, Shenzhen, China*
[4] heyejun@ieee.org

***Abstract*—Packet-level generative models are designed to be employed for evaluating and designing high layer wireless communications protocols. The significance of these models arises from their ability to replace the entire physical communication layer, hence reducing the time and complexity of wireless system simulations. In this paper, an enhanced general packet radio service (EGPRS) transmission system is utilized to obtain reference bursty packet error sequences. We then propose a new hidden Markov model with two processes to generate packet error sequences of any desired length. The attained burst error statistics demonstrate a satisfactory performance compared to those of reference packet error sequences.**

I. Introduction

A digital channel consists of the complete transmission chain in a communication system, including the transmitter, the physical channel, and the receiver. The erroneous packets encountered in the digital channels are correlated. Error models have been proposed to describe the packet error sequences by using the error burst statistics. Error models are either descriptive or generative [1]. Descriptive models statistically characterise target error sequences obtained from real systems. Generative models are an alternative mechanism that can produce error sequences having similar statistics to those of general models. Simulating real systems is often time-consuming and therefore, generative models are an alternative solution. Packet-level error models are essential for evaluating high layer wireless communication protocols, e.g., in the media access control (MAC) layer [2].

There has been extensive research on bit-level generative models, which can generally be divided into five classes. These include Markov models [3], [4] such as the well-known simplified Fritchman model (SFM) [5], hidden Markov models (HMMs) such as using the Baum-Welch (BW) algorithm [6] and double embedded processes [7], stochastic-context free grammars models [8], chaotic models [9], and the deterministic process based generative models (DPBGMs) [10] which outperform the first four classes of generative models in terms of fitting the desired burst error statistics. However, DPBGMs do not create 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. This restricts their ability to adaptively generate error sequences when the channel conditions vary. In principle, all the five classes of bit-level generative models can be adjusted to generate packet error sequences by properly tuning the involved parameters. The purpose of this paper is to follow the line of the bit-level double embedded processes based HMM (DEPHMM) we proposed in [7] and extend to a packet-level DEPHMM for generating packet error sequences. Also, we will compare the newly developed packet-level DEPHMM with the packet-level DPBGM and SFM in [11].

This paper is organized as follows. Section II introduces some terms and definitions related to packet error sequences and their statistics. Section III demonstrates the structure and parameterization of the packet-level DEPHMM. In Section IV, the performance of the developed DEPHMM is shown through simulations. Conclusions are drawn in Section V.

II. Terms and Definitions

In this section we clarify the terms and relevant burst error statistics we use in this paper for binary packet error sequences. An erroneous or failed packet is represented by "1", whereas a correctly received packet is represented by "0".

A *gap* is described as a series of consecutive zeros between two ones. Its length equals the number of zeros. An *error cluster* is a series of consecutive ones having a length equal to the number of ones. An *error-free burst* is a sequence of consecutive zeros that have at least η bits in length. An *error burst* is a sequence of zeros and ones delimited by two ones. It is separated from other error bursts by error-free bursts.

In order to analyze packet-level error sequences, many burst error statistics have been elaborated in the literature [11]. These statistics are also used to scrutinize the performance of generated error sequences. The burst error statistics that will be used in this paper are listed below [11]:

1) $P(0^{m_0}|1)$: error-free run distribution (EFRD), which is the probability that an error packet is followed by m_0 or more error-free packets.
2) $P_{EB}(m_e)$: error-burst distribution (EBD), which is the cumulative distribution function (CDF) of error burst lengths m_e.

978-1-4244-2721-5/09/$25.00 ©2009 IEEE

3) $P(1^{m_c}|0)$: error-cluster distribution (ECD), which is the probability that a zero is followed by at least m_c successive ones.
4) $\rho(\Delta k)$: packet error correlation function (PECF), which is the conditional probability that the Δkth packet following an error packet is also in error.
5) $P(m,n)$: block error probability distribution (BEPD), which is the probability that m out of n packets are in error.

III. The proposed Generative Model

The illustration of the packet DEPHMM is depicted in Fig. 1. The first step in formulating the DEPHMM is to extract the error bursts from a reference error sequence. For the first process, only one state (**0**) is sufficient to represent the error-free bursts because it corresponds to zeros only. Whereas, the error bursts have ones and zeros organised differently. Therefore, error bursts deserve partitioning into many groups (**1, ...,** N). Each group should convey a shared structural behavior. Because the errors mainly govern the behavior of the packet error sequence, we decided to use the number of error packets in each error burst as a criterion for partitioning. However, the length of the error bursts varies and this would affect the number of error packets. Therefore, the ratio of the number of error packets in an error burst to the length of the error burst is the best choice for the purpose of partitioning. Consequently, the range of ratio values should be partitioned to accommodate the number of error burst states, where each state has approximately the same numbers of error bursts.

The process of creating error-free burst lengths is straightforward. Error-free burst lengths can be created by utilizing the reference error-free burst lengths distribution. The real challenge now is to create error burst lengths. Since our major concern is to detail the construction of errors within an error burst, we dedicate several substates to the error cluster lengths of each class of the first process and a single substate for the gap lengths of the error bursts. Each error cluster substate represents a single error cluster length. Similar to the error-free bursts, the generation of gap lengths within the error bursts of each class depends on their length probability distribution. Combining the error cluster substates with the gaps substate is the task of the second process, whereas the first process is dedicated to combining the error bursts with the error-free bursts.

The parameters of the DEPHMM are as follows:

1) N: the number of states for error bursts, i.e., $S = \{s_1, s_2, ..., s_N, s_{N+1}\}$, where S is the set of states. The parameter N is selected according to the desired accuracy.
2) M_u: the number of error cluster substates in each state, i.e., $V_u = \{v_{1u}, v_{2u}, ..., v_{Mu}, v_{Mu+1}\}$, where V_u is the set of substates and $u = 1, ..., N$. The parameter M_u is set to the number of error cluster lengths in each error burst state.
3) $\mathbf{F} = (f_{i,j})$: the state transition matrix, where $f_{i,j}$ is the transition probability from s_i to s_j, such that

$$f_{i,j} = P\left[Q_{t+1} = s_j \mid Q_t = s_i\right], 1 \le i, j \le N+1$$
$$= \begin{cases} 1, & 1 \le i \le N, j = N+1, \\ \frac{N_{EB,j}}{\sum_{j=1}^{N} N_{EB,j}} \approx \frac{N_{EB,j}}{N \times N_{EB,N}}, & i = N+1, 1 \le j \le N, \\ 0, & \text{otherwise}, \end{cases} \quad (1)$$

with Q_t being the current state at time t and $N_{EB,j}$ the number of error bursts in s_j.
4) $\mathbf{D}_u = ((d_{h,k})_u)$: the substate transition matrix, where $(d_{h,k})_u$ is the transition probability from v_{h_u} to v_{k_u}, such that

$$(d_{h,k})_u = P\left[R_{t+1} = v_{k_u} | R_t = v_{h_u}\right], 1 \le h_u, k_u \le M_u + 1$$
$$= \begin{cases} 1, & 1 \le h_u \le M_u, k_u = M_u + 1, \\ \frac{N_{C,k_u}}{\sum_{k_u=1}^{M_u} N_{C,k_u}}, & h_u = M_u + 1, 1 \le k_u \le M_u, \\ 0, & \text{otherwise}, \end{cases} \quad (2)$$

with R_t being the current state at time t and N_{C,k_u} the number of error clusters in k_u.
5) $\mathbf{A} = (a_j(n))$: the first process emission probability distribution matrix, where $a_j(n)$ $(1 \le j \le N+1)$ is the probability of getting the burst y_n in state s_j, that is

$$a_j(n) = P\left[y_n \text{ at } t | Q_t = s_j\right], 1 \le n \le N_{EB,j}, N_{EFB,j}.$$

$N_{EFB,j}$ is the number of error-free bursts in s_j.
6) $\mathbf{B} = (b_{k_u}(m))$: the second process gap emission probability distribution matrix, where $b_{k_u}(m)$ $(k_u = M_u + 1)$ is the probability of getting the gap length x_m in state v_{k_u}, that is

$$b_{k_u} = P\left[x_m \text{ at } t | R_t = v_{k_u}\right], 1 \le m \le N_{G,k_u}.$$

N_{G,k_u} is the number of gaps in k_u.
7) $\mathbf{\Pi}_u = ((\pi_k)_u)$: the initial substate distribution vector, where $(\pi_k)_u$ is the probability of v_{k_u} to be an initial substate.

$$\mathbf{\Pi}_u = \left(d_{M_u+1,1}, ..., d_{M_u+1,M_u}, 0\right), \quad (3)$$

which is important because it assures the initiation of an error burst by an error cluster. Similarly, we can obtain the termination substate distribution vector $\mathbf{\Omega}_u$, which assures that the error burst is ending with an error cluster, and the initial state distribution vector.
8) $\delta_{n,u}$: error burst length values obtained from the reference error burst length distribution. These values regulate the termination of error burst generation, so that Ω_u shall be activated according to them. The activation takes place when the generated error burst length becomes either equal or around a chosen δ_n. The deviation from δ_n should be small enough, otherwise, the current generated error burst will be discarded.
9) Γ: generated packet error sequence length. This value terminates the packet error sequence generation once it is reached or exceeded.

These parameters construct the packet DEPHMM and explain its algorithm. Note that the percentage of the generated error packets among the total generated packets for each state should approximately match that of the corresponding state of the reference error sequence. If the difference is large, then the error burst state should be further partitioned.

IV. Simulation Results and Discussions

In order to obtain reference packet error sequences for the purpose of parametrization, we simulated a coded EGPRS transmission system, as detailed in [11]. It consists of a convolutional encoder/decoder, a burst interleaver/deinterleaver, a Gaussian minimum shift keying (GMSK) modulator/demodulator, a Viterbi equalizer, and a cyclic redundancy check (CRC) for error detection. The underlying channel is tailored to a typical urban (TU) environment with a mobile speed of 3 km/h. In this paper, we use an example reference packet error sequence with a carrier-to-interference ratio (CIR) of 11 dB and length of 1 million. The packet error sequence is structured by allocating a one for a failed or erroneous packet which contains at least one undecodable error, and a zero for a correctly decoded packet. The value of η is chosen to be 50 from the EFRD plateau and $\Gamma = 1.7$ million bits. The error bursts can then be extracted and partitioned into classes. The number of partitions should be high, e.g., more than 10. This number affects the accuracy of the final results and the complexity of the model [7]. In this paper, $N = 20$ is used. For the purpose of comparisons, a packet-level DPBGM and SFM [11] were implemented. The parameters of the packet-level DPBGM are $R_B = 0.052$ and $q_s = 0.01$. Subsequently, the vector $\Psi = (9, 10, 0.09, 0.2255, 300.85Hz, 0.0394ms)$, and the number of states used for the SFM is 6 [11].

In order to evaluate any generative model, we have to find out how closely its burst error statistics match those of the descriptive model. Figs. 2–6 demonstrate the performance of the packet-level DEPHMM, DPBGM, and SFM compared with the descriptive model in terms of the burst error statistics mentioned in Section II. The EFRDs and EBDs are shown in Figs. 2 and 3, respectively, where the DEPHMM has negligible degradation from the descriptive model. In Fig. 4, the ECD of the DEPHMM has a little divergence from the DEPHMM for long error clusters with lengths not less than 8. This is because the probability of selecting these lengths are smaller than others in the second process of the DEPHMM. In Fig. 5, the PECF of the DEPHMM is shown. It has a sufficient image of the descriptive model at high correlation values, but it fails to mimic the distinct breakpoints. Finally, The BEPD is illustrated in Fig. 6, where the value of $n = 55$ was selected. This statistic shows a good match between the DEPHMM and the descriptive model. Note that the SFM fails to match most of the burst error statistics except the EFRD. It is clear that the DEPHMM burst error statistics have small discrepancies from the DPBGM statistics and the descriptive model. In fact, the DPBGM retrieves the error bursts from the reference packet error sequence rather than constructing them by itself. This explains the high quality performance of the DPBGM. Nevertheless, this process of borrowing error bursts from the reference sequence is not desirable for the development of future adaptive generative models.

V. Conclusions

The DEPHMM is capable of constructing binary packet error sequences having burst error statistics that match closely those of reference packet error sequences derived from an EGPRS system. This is achieved by forming detailed error bursts through the second process of the DEPHMM. The DPBGM can approximate the desired burst error statistics of the descriptive model very well. However, it does not create error bursts by itself, which is not desirable for the application to adaptive general modelling. The SFM failed to match most of the burst error statistics of the descriptive model.

Acknowledgment

Omar S. Salih, Cheng-Xiang Wang, and David I. Laurenson would like to acknowledge gratefully the sponsorship of this work by the EPSRC and Philips Research Cambridge, UK, the support from the Scottish Funding Council for the Joint Research Institute in Signal and Image Processing between the University of Edinburgh and Heriot-Watt University, and thank Mr. Rob Davies from Philips Research Cambridge for helpful discussions. Yejun He acknowledges the support of his work by National Natural Science Foundation of China (Grant No.60972037) and the Fundamental Research Program of Shenzhen City (Grant No. JC200903120101A).

References

[1] L. N. Kanal and A. R. K. Sastry, "Models for channels with memory and their applications to error control," *Proc. of the IEEE*, vol. 66, no. 7, pp.724–744, July 1978.

[2] A. Konard, B. Y. Zhao, A. D. Joseph, and R. Ludwig, "A Markov-based channel model algorithm for wireless networks," in *Proc. of 4th ACM International Workshop on Modeling, Analysis and Simulation of Wireless and Mobile Systems.*, Rome, Italy, July 2001.

[3] E.O. Elliot, "Estimates of error rates for codes on burst-noise channels," *Bell Syst. Tech. J.*, vol. 42, pp. 1977–1997, Sept. 1963.

[4] X. Long, X. Gan, Y. Xu, J. Liu; M. Tao, "An estimation algorithm of channel state transition probabilities for cognitive radio systems," *Proc. CrownCom'08*, Singapore, May 2008, pp. 1–4.

[5] B. D. Fritchman, "A binary channel characterization using partitioned Markov chains," *IEEE Trans. Information Theory*, vol. 13, no. 2, pp. 221–227, Apr. 1967.

[6] 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.

[7] O. S. Salih, C.-X. Wang, D. I. Laurenson,"Double embedded processes based hidden Markov models for binary digital wireless channels," *Proc. ISWCS'08*, Reykjavik, Iceland, Oct. 2008, pp. 219–223.

[8] 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.

[9] E. Costamagna, L. Favalli, P. Savazzi, F. Tarantola, "Evaluation of frame-bit error sequence generators at the output of the decoding process," in *Proc. IEEE VTC'04-Spring*, Milan, Italy, May. 2004, pp. 1028–1032.

[10] 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.

[11] C. X. Wang and M. Pätzold, "Packet-level error models for digital wireless channels," *Proc. IEEE ICC'05*, Seoul, Korea, May 2005, pp. 2184–2189.

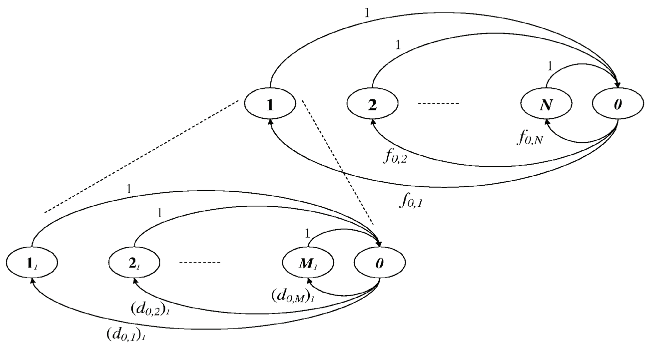

Fig. 1. The packet-level DEPHMM.

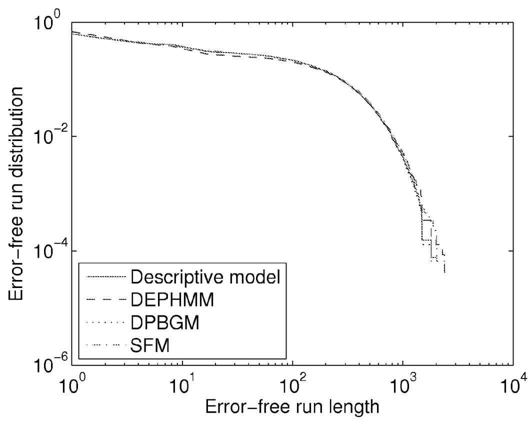

Fig. 2. The EFRDs of the descriptive model obtained from the EGPRS system and different packet-level generative models.

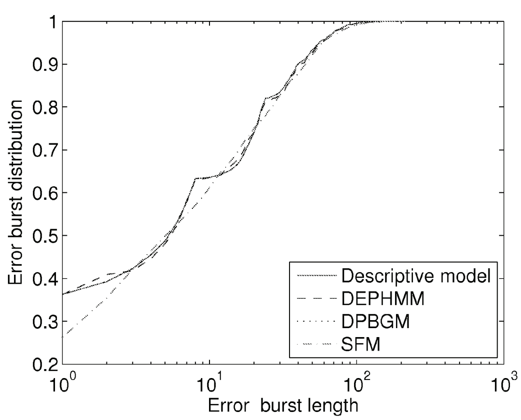

Fig. 3. The EBDs of the descriptive model obtained from the EGPRS system and different packet-level generative models.

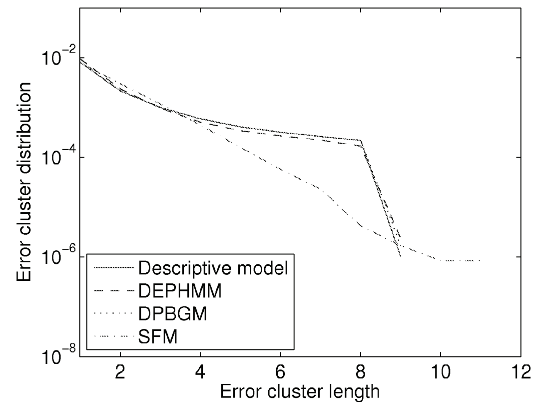

Fig. 4. The ECDs of the descriptive model obtained from the EGPRS system and different packet-level generative models.

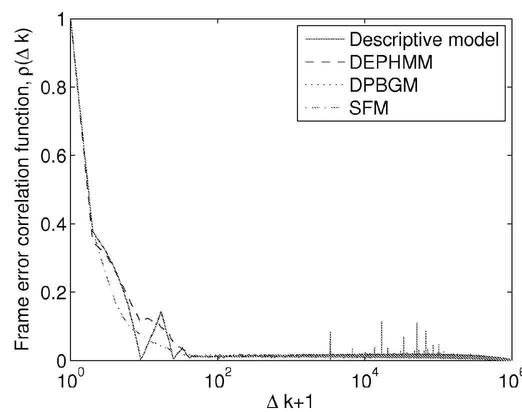

Fig. 5. The PECFs of the descriptive model obtained from the EGPRS system and different packet-level generative models.

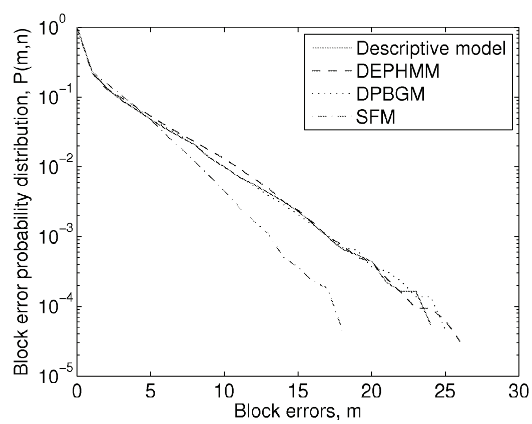

Fig. 6. The BEPDs of the descriptive model obtained from the EGPRS system and different packet-level generative models.