Error Models for Digital Channels and Applications to Wireless Communication Systems

by

Omar S. Salih

A thesis submitted in partial fulfilment for the degree of
Doctor of Philosophy

at

Heriot-Watt University
School of Engineering and Physical Sciences

July 2013

Abstract

Digital wireless channels are extremely prone to errors that appear in bursts or clusters. Error models characterise the statistical behaviour of bursty profiles derived from digital wireless channels. Generative error models also utilise those bursty profiles in order to create alternatives, which are more efficient for experimental purposes. Error models have a tremendous value for wireless systems. They are useful for the design and performance evaluation of error control schemes, in addition to higher layer protocols in which the statistical properties of the bursty profiles are greatly functional. Furthermore, underlying wireless digital channels can be substituted by generated error profiles. Consequently, computational load and simulation time can be significantly reduced when executing experiments and performing evaluation simulations for higher layer communications protocols and error control strategies.

The burst error statistics are the characterisation metrics of error models. These statistics include: error-free run distribution; error-free burst distribution; error burst distribution; error cluster distribution; gap distribution; block error probability distribution; block burst probability distribution; bit error correlation function; normalised covariance function; gap correlation function; and multigap distribution. These burst error statistics scrutinise the error models and differentiate between them, with regards to accuracy. Moreover, some of them are advantageous for the design of digital components in wireless communication systems.

This PhD thesis aims to develop accurate and efficient error models and to find applications for them. A thorough investigation has been conducted on the burst error statistics. A breakdown of this thesis is presented as follows.

Firstly, an understanding of the different types of generative error models, namely, Markovian based generative models, context-free grammars based generative models, chaotic models, and deterministic process based generative models, has been presented. The most widely used models amongst the generative models have been compared with each other consulting the majority of burst error statistics. In order to study generative error models, error burst profiles were obtained mainly from the Enhanced General Packet Radio Service (EGPRS) system and also the Long Term Evolution (LTE) system.

Secondly, more accurate and efficient generative error models have been proposed. Double embedded processes based hidden Markov model and three-layered processes

based hidden Markov model have been developed. The two types of error profiles, particularly the bit-level and packet-level error profiles were considered.

Thirdly, the deterministic process based generative models' parameters have been tuned or modified in order to generate packet error sequences rather than only bit error sequences. Moreover, a modification procedure has been introduced to the same models to enhance their generation process and to make them more desirable.

Fourthly, adaptive generative error models have been built in order to accommodate widely used generative error models to different digital wireless channels with different channel conditions. Only a few reference error profiles have been required in order to produce additional error profiles in various conditions that are beneficial for the design and performance evaluation of error control schemes and higher layer protocols.

Finally, the impact of the Hybrid Automatic Repeat reQuest (HARQ) on the burst error statistics of physical layer error profiles has been studied. Moreover, a model that can generate predicted error sequences with burst error statistics similar to those of error profiles when HARQ is included has been proposed. This model is constructive in predicting the behaviour of the HARQ in terms of a set of higher order statistics rather than only predicting a first order statistic. Moreover, the whole physical layer is replaced by adaptively generated error profiles in order to check the performance of the HARQ protocol.

The developed generative error models as well as the developed adaptive generative error models are expected to benefit future research towards the testing of many digital components in the physical layer as well as the wireless protocols of the link and transport layers for many existing and emerging systems in the field of wireless communications.

To my parents, wife, and daughter

Acknowledgements

All praise and thanks to Allah, The Lord of the Worlds, The Most Beneficent, The Most Merciful. I would like to express my utmost gratitude to my Lord, Allah, then to my supervisors, colleagues, friends, and family for their assistance in order to accomplish this thesis.

My greatest appreciation goes to my main supervisor, Prof. Cheng-Xiang Wang. I thank him for the guidance, encouragement, patience, efforts, and advice he has provided throughout my PhD study. I'm also grateful to my second supervisor, Dr. Dave Laurenson who has helped me with admirable wisdom and depth as well as providing me with assuring words. I owe to both my supervisors a massive thank you. More appreciations go to my industrial supervisors, namely Dr. Tim Mousley and Mr. Rob Davies, for their generous support, conversations, and advice.

I also gratefully acknowledge the sponsorship of this work by the EPSRC and Philips Research Cambridge, UK, as well as 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, as a part of the Edinburgh Research Partnership in Engineering and Mathematics (ERPem).

I am enormously delighted to have the privilege of knowing my colleagues of past and present, Xuemin Hong, Xiang Cheng, Zengmao Chen, Ivan Ku, Margaret Anyaegbu, Raul Hernandez, Yi Yuan, Ammar Ghazal, Yu Fu, Fourat Haider, and Murtadha Al-Saedy. I thank you for all the wonderful moments and may our friendship lasts a lifetime.

Last but not least, my special gratitude goes to my wonderful parents, brothers, sisters, wife, and daughter for their unconditional love and unwavering support.

Contents

List of Figures

List of Tables

Abbreviations

3GPP	Third Generation Partnership Project
3LHHM	Three Layered Hidden Markov Model
4G	Fourth Generation
ABWHMM	Adaptive Baum-Welch based Hidden Markov Models
ADF	Average Duration of Fades
ADIF	Average Duration of Inter-Fades
ADPBGM	Adaptive Deterministic Process Based Generative Model
ADPB-HMM	Adaptive Deterministic Process Based Hidden Markov Model or Adaptive DPB-HMM
AMC	Adaptive Modulation and Coding
ARQ	Automatic Repeat reQuest
ASFM	Adaptive Simplified Fritchman's Model
AWGN	Additive White Gaussian Noise
BBPD	Block Burst Probability distribution
BCF	Bit correlation function
BEPD	Block Error Probability distribution
BER	Bit Error Rate
BS	Base Station
BSC	Binary Symmetric Channel
BW	Baum-Welch algorithm
BWHMM	Baum-Welch based Hidden Markov Models
CAGM	Chaotic Attractor Generative Model
CDF	Cumulative distribution Function
CIR	Carrier-to-Interference Ratio

CQI	Channel Quality Indicator
CRC	Cyclic Redundancy Check
DEPHMM	Double Embedded Processes based Hidden Markov Model
DL-SCH	Downlink Shared Channel
DPBGM	Deterministic Process Based Generative Model
DPB-HMM	Deterministic Process Based Hidden Markov Model
DVB-T	Digital Video Broadcasting – Terrestrial
EB	Error Burst
EC	Error Cluster
ECD	Error Cluster Distribution
EESM	Exponential Effective SIR Mapping
EFB	Error-Free Burst
EFRD	Error-Free Run Distribution
EG	Error Gap (gap within an EC length)
EGPRS	Enhanced General Packet Radio Service
FEC	Forward Error Control
FER	Frame Error Rate
G	Gap
GCF	Gap Correlation Function
GD	Gap Distribution
GMSK	Gaussian Minimum Shift Keying
GPS	Global Position System
HARQ	Hyprid Automatic Repeat reQuest
HF	High Frequency
HMM	Hidden Markov Model
IFH	Ideal Frequency Hopping
ISI	Intersymbol Interference
ITU	International Telecommunication Union
LCR	Level Crossing Rate
LTE	Long Term Evolution
MAC	Media Access Control

MGD	Multi-Gap Distribution
MIMO	Multiple Input Multiple Output
MISO	Multiple Input Single Output
MM	Markov Models
NCF	Normalised Covariance Function
NIFH	Non ideal Frequency Hopping
OFDM	Orthogonal Frequency Division Multiplexing
OFDMA	Orthogonal Frequency Division Multiple Access
PedA	Pedestrian A
PedB	Pedestrian B
PER	Packet Error Rate
PDF	Probability Density Function
PECF	Packet Error Correlation Function
PEPD	Packet Error Probability Distribution
PNE	Peak Number of Errors
QoS	Quality of Service
RA	Rural Area
RF	Radio Frequency
Rx	Receiver
SBECF	Soft Bit-Error Correlation Function
SBEPD	Soft Block Error Probability Distribution
SCFG	Stochastic Context-Free Generative model
SCM	Spatial Channel Model
SDSD	Soft Decision-Symbol Distribution
SECD	Soft Error-Cluster Distribution
SEFRD	Soft Error-Free Run Distribution
SFM	Simplified Fritchman's Model
SGD	Soft Gap Distribution
SINR	Signal-to-Interference-plus-Noise Ratio
SIR	Signal-to-Interference Ratio
SISO	Single Input Single Output

SNR	Signal-to-Noise Ratio
TB	Transport Block
TDMA	Time Division Multiple Access
TU	Typical Urban
TV	Television
Tx	Transmitter
UE	User Equipment
UHF	Ultra High Frequency
UWB	Ultra Wide Band
VHF	Very High Frequency

Symbols

$\lfloor \cdot \rfloor$	the floor function
$\lceil \cdot \rceil$	the ceiling function
$\mathbf{A}$	the first process emission probability distribution matrix in DEPHMM
$\mathcal{A}$	the first layer emission probability distribution matrix in 3LHMM
$a_j(n)$	the probability of getting the burst y_n in State s_j
$\mathbf{B}$	the second process gap emission probability distribution matrix in DEPHMM
$\mathcal{B}$	the second layer emission probability distribution matrix in 3LHMM
$b_{k_u}(m)$	the probability of getting the gap x_m in State v_{k_u}
$c_{i,n}$	the sum of sinusoids gains
$Cov(l)$	the normalised covariance function
D_k	the descending state of the SCFG
$\mathbf{D}_u$	the substate transition matrix in DEPHMM
$(d_{h,k})_u$	the transition probability from Substate v_{h_u} to Substate v_{k_u}
$\mathcal{E}$	the third layer gap emission probability distribution matrix in 3LHMM
$\mathbf{EB}_{rec}$	the error burst lengths record

$\widetilde{\mathbf{EB}}_{rec}$	the error burst length generator
$\mathbf{EC}_i$	the error cluster recorder of the reference error sequence
$\mathbf{EC}_{i,HARQ}$	the new error cluster recorder after imposing HARQ
$\mathbf{EC}_{i,h}$	the new error cluster lengths inside each $\mathbf{EC}_i$ after imposing HARQ
$\mathbf{ECG}_j$	the error cluster and gap record for each error burst
$\widetilde{\mathbf{ECG}}_j$	the error cluster and gap generator
$\mathbf{EFB}_{rec}$	the error-free burst lengths record
$\widetilde{\mathbf{EFB}}_{rec}$	the error-free burst length generator
$\mathbf{EG}_{i,u}$	the new gap lengths inside $\mathbf{EC}_i$ after imposing HARQ
e_j	the soft decision symbols
$e_{k_{p,q}}(\ell)$	the probability of getting the gap z_ℓ in Substate $v_{k_{p,q}}$
$\mathbf{F}$	the state transition matrix in DEPHMM
F_s	the transmission rate
$f_{i,j}$	the transition probability from State s_i to State s_j
$f_{i,n}$	discrete frequencies of the fading process
$f(\cdot)$	exponential or polynomial function
G_j	the gap lengths recorder of each error burst
$G(m_g)$	the gap distribution
i	general index
j	general index
k	general index or nonnegative integer
$L_{p,q}$	the number of error clusters substates in each internal state in 3LHMM
l_i	the number of new error clusters in each $\mathbf{EC}_i$ after imposing HARQ
ln	the natural algorithm
M_{EB}	the mean value of error-free burst lengths
M_{EFB}	the mean value of error burst lengths

M_u	the number of error cluster substates in DEPHMM
$m, \mathpzc{m}$	general indices
m_+	the number of nonnegative integers
m_-	the number of negative integers
m_0	the length of error-free digits
m_c	the length of consecutive erroneous digits
m_e	the error burst length
$m_{\bar{e}}$	the error-free burst length
m_g	the gap length
N	the number of states in Markov models
$\mathscr{N}$	the number of subcarriers in OFDM
$\mathcal{N}(\cdot)$	the normal distribution
N_{C,k_u}	the number of error clusters in Substate v_{k_u}
$\mathcal{N}_{EB}$	the total number of error bursts
$\tilde{\mathcal{N}}_{EB}$	the total number of generated error bursts
$N_{EB,j}$	the number of error bursts in State s_j
$\mathcal{N}_{EFB}$	the total number of error-free bursts
$\tilde{\mathcal{N}}_{EFB}$	the total number of generated error-free bursts
N_p	the packet size in bits
N_s	the number of states for soft error bursts
N_t	the length of the reference error sequence
$\tilde{N}_t$	the required length for the generated error sequence
$\tilde{N}_\zeta(r_{th})$	the LCR at a specified threshold of the fading process
NEL	the number of errors in a block of length L in an error burst
$n, \mathpzc{n}$	general indices
p	general index
$P(0^{m_0}\|1)$	the error-free run distribution
$P(1^{m_c}\|0)$	the error cluster distribution
$P_{EB}(m_e)$	the error burst distribution

$P_{EFB}(m_{\bar{e}})$	the error-free burst distribution
$P(e_j)$	the soft decision-symbol distribution
$P(m_+)$	the soft error-free run distribution
$P(m_-)$	the soft error-cluster distribution
$P(m,n)$	the block error probability distribution
q	general index
q_s	a very small quantity that determine the
	maximum measurement error of the LCR
Q_t	the current state at time t
$Q(l,n)$	the block burst probability distribution
R_t	the current substate at time t in DEPHMM
$\mathcal{R}_B$	the ratio of M_{EB} and M_{EFB}
$\tilde{\mathcal{R}}_B$	the ratio of the ADF and ADIF
r_{th}	the fading process threshold
S	the set of states
$S[\cdot]$	the shaping function of the CAGMs
T_t	the total transmission time
$\tilde{T}_t$	the required time to transmit a certain length of bits
$\tilde{T}_{\zeta_-}(r_{th})$	the ADF
$\tilde{T}_{\zeta_+}(r_{th})$	the ADIF
T_A	the sampling time
t	time
U_k	the ascending state of the SCFG
X,Y	error sequences at certain SNR
Z	the required error sequences at certain SNR
α,β	the weighting parameters of the adaptive generative models
Γ	the generated error sequence length in DEPHMM
γ	the optimisation parameter in EESM
Δk	the length of certain number of bits or packets
δ_{eff}	the instantanous effective SNR
$\delta_{n,u}$	the error burst lengths in error states of DEPHMM

ϵ	the mean value of the exponential distribution
ζ	a positive integer number
$\tilde{\zeta}(\cdot)$	the fading process
η	the minimum error-free burst length
η_i	the initial or minimum value of η
η_f	the final or maximum value of η
$\theta_{i,n}$	the uniformly distributed phases for the fading process
λ	the parameter vector of the BWHMM
μ	the mean value of the Gaussian distribution
$\tilde{\mu}_i$	the sum of sinusoids
$\mathbf{\Pi}_u$	the initial substate distribution vector in DEPHMM
$(\pi_k)_u$	the probability of v_{k_u} to be an initial substate in DEPHMM
$\rho(\Delta k)$	the bit (or packet) error correlation function
σ	the standard deviation
σ_0	the square root of the mean power of the sum of sinusoids signal
Ψ	the parameter vector of the DPBGM
ψ	a subset vector of Ψ
$\mathbf{\Omega}_u$	the termination substate distribution vector in DEPHMM
$(\omega_k)_u$	the probability of v_{k_u} to be the final substate

Chapter 1

Introduction

1.1 Problem Statement

When we look around us, we see many devices that are receiving instructions and are communicating with each other. However, we cannot see how these devices can maintain contact with each other. It is true that wireless communication is now very much part of our lives which we cannot relinquish. In fact, wireless communication makes our life fruitful as it is able to convert the whole world into a small village whereby its residents figure as a family. Wireless communication has also an immense importance for spreading knowledge within a global context. It has assisted in rapidly developing the world as information has become like an open source from where it can be acquired by anyone. Hence, developments are built up upon what has already been achieved and the experiences of others .

Wireless communications have been receiving a numerous number of applications, especially when wired communication is not possible or is very expensive to implement. The mission of the curiosity rover which landed on Mars and the jump of Felix Baumgartner from the edge of space would have been impossible without wireless communication. Additional applications include: satellite communications; global positioning system (GPS); TV and radio broadcasting; cellular communications; marine communications; aeroplanes and traffic control systems; bus waiting time monitoring; internet broadcasting; ultrasonic and infrared remote controlling; computer peripherals communications; wireless home entertainment systems; and much more applications can be related to defence, education, and health.

Investigating the history of wireless communication, Alexander Bell and Charles Tainter made the first wireless telephone conversation in 1880 whereby the signals were modulated over light beams. However, their invention did not receive applications at that time due to the lack of understanding of optical communications systems. In 1879, David Hughes transmitted radio signals over a few hundred yards utilising a clockwork keyed transmitter. In 1885, Thomas Edison used a vibrator magnet for induction transmission. Later on in 1888, Edison deployed a system of signalling on the Lehigh Valley Railroad; and in the same year, Heinrich Hertz demonstrated the existence of electromagnetic waves, which are the underlying basis of most wireless technologies. The theory of electromagnetic waves was already predicted by James Maxwell and Michael Faraday. However, Hertz demonstrated that electromagnetic waves could be transmitted and received by experimental means. In the late 1880s Jagadish Bose developed a wireless detector and enhanced the knowledge of millimetre-length electromagnetic waves. Nikola Tesla was later engaged with practical applications of wireless radio communication for short ranges. In 1901, a huge improvement to wireless communications occurred when Guglielmo Marconi transmitted the first radio signal across the Atlantic. In 1909 both Guglielmo Marconi and Karl Ferdinand Braun were awarded the Nobel Prize for Physics for their contribution towards wireless telegraphy [1].

Any wireless communication system must include a transmitter and receiver or a transceiver in order to send informations from one side and to be detectable at the other side. The problem is that some of the detected information may be received differently from that which has been sent. This is due to the errors which are occurring between the transmitter and the receiver. It is very important to study these errors from different perspectives. One perspective is to understand the sources or the causes of errors in the transmission channel. Another perspective is to obtain knowledge about the statistical properties of these errors in terms of numbers and distributions. For both perspectives the final goal is to tackle these errors by suspending or mitigating their effects on the received signals. Some of the main causes of error events are [2]:

- Attenuation: which is a decrease in the electromagnetic energy at the receiver due to long distance propagation and the existence of obstacles between the transmitter and the receiver.
- Intersymbol interference (ISI): symbols are interfering with each other resulting in partial cancellation of some symbols due to delays in receiving them.
- Doppler shift: which is a frequency shift in the arriving signal because of the relative velocities of the transmitter and the receiver.
- Multipath fading: which is a fluctuation in the amplitude, phase, and angle of the signal received at the receiver.
- Internal processing of symbols: these are errors occurred inside the processing parts of symbols inside the transceiver, such as the errors that happen at the demodulation because of the decision scheme.

The main factors that influence the wireless radio propagation are illustrated in Figure 1.1 to be:

- Reflection: occurs when an electromagnetic wave impinges on a smooth surface that has a larger size than the wavelength of the radio wave. Reflected signals are collected constructively or destructively at the receiver.
- Diffraction: occurs when the path of the electromagnetic wave hits an impenetrable object that has larger dimensions than the signal wavelength. Subsequently, secondary waves appear behind the obstructing object without any line of sight (LOS) path between them. Therefore, signals can be detected by the receiver despite the absence of LOS path.

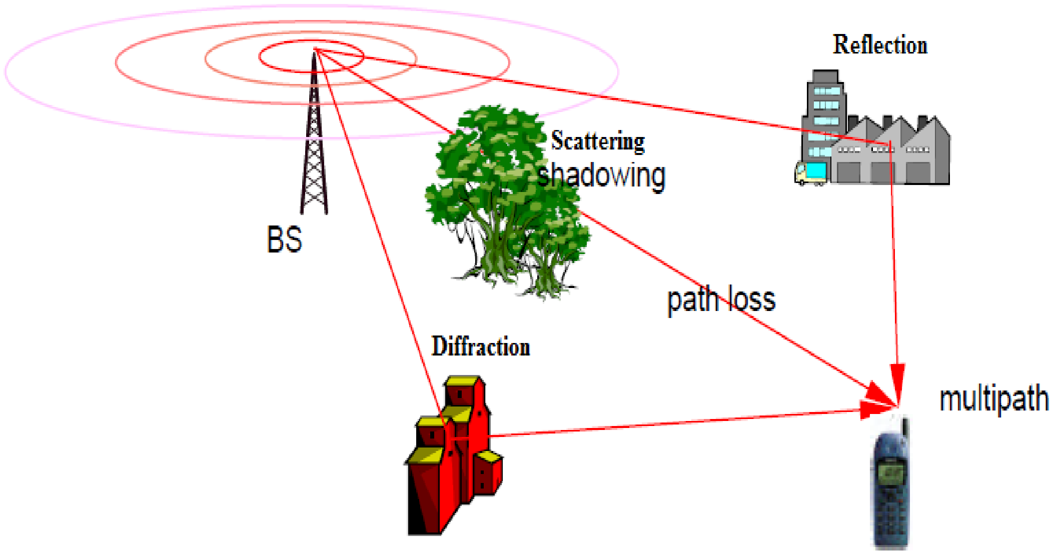

FIGURE 1.1: Reflection, diffraction, and scattering are the main factors of wireless propagation fading [3].

Diffraction can also be called shadowing because the diffracted signals can arrive at the receiver even when shadowed by an impenetrable obstruction.

- Scattering: occurs when the radio signals meet obstructing objects of dimensions that are on the order (or less) of the electromagnetic wavelength. Scattering allow the signals to be emitted in many different directions. Foliage, lamp posts, and street signs are typical objects that cause scattering.

1.2 Motivation

The engineers and designers of wireless communication systems must be aware of the errors, which are encountered in the wireless channel, in order to enhance the quality of service that is provided to customers. Better communication links can be achieved by deploying appropriate modulation, coding schemes, interleavers, and other channel components as well as high layer protocols, such as the Automatic Repeat reQuest (ARQ). Error models for digital wireless channels can identify such errors and analyse them statistically, in order to better understand the impairments inside the digital wireless channels and to work thoroughly to mitigate the impact of such errors on the quality of the received information. A reliable wireless communication really needs a thorough understanding of errors in the wireless channels

An accurate and comprehensive reproducible error model is in high demand. Such accurate models assist the wireless communication engineers in evaluating the error control schemes and high layer protocols and algorithms in a controlled and repeatable way. This could have a significant impact on research and the industry pertaining to wireless communication as error models considerably reduce the computational load and simulation time of executing a wireless communication system when an evaluation or a design of a component is in progress.

Thus imagine that you are testing a wireless protocol and this protocol has many variations with regards to its parameters or steps. Each time you test one parameter you would need to set up the equipment again and send a long sequence of data and wait until you had received all of the data, which could take hours if not days. However, the underlying wireless channels could be replaced by an error model that could generate data profiles of tremendous lengths as required within seconds in order to examine the associated component or protocol.

It is also deemed extremely desirable to attain error profiles for various channel conditions or different digital wireless channels from a few already obtained error profiles at fixed channel conditions. There is no doubt that this idea leverages the reducibility of error models and enriches their applications.

The statistical characteristics for higher layer protocols, particularly the ARQ, occupy part of our interests. Understanding these characteristics takes us to the core of our purpose, which is reducing the computational burden by simplifying the simulation model; and hence saving our valuable time.

It is extremely enviable for researchers and designers to have error models which are accurate, as accuracy has an impact on the final decision of the designation or evaluation. However, any accurate model requires complexity in computation. In this thesis, we have put our efforts into designing accurate error models whilst keeping the computation complexity acceptable. Thus, the parametrisation of our models are considered to be simple and easy to understand and apply. Moreover, for some models that have certain disadvantages which affect their production ability, we have proposed a solution for alleviating their problems.

1.3 Contributions

The key contributions of the thesis are summarised as follows:

- **Modelling the digital wireless channels:**

  Models of digital channels are to be proposed based on the idea of Hidden Markov Models (HMMs). The first model has two embedded processes whereby the first process is dedicated to assembling error bursts with error-free bursts. However, the second one is devoted to creating individual error bursts whilst employing the maximum gap norm within error bursts. This model is not only applicable for hard bit error profiles but also for soft bit error profiles as well as packet error profiles.

  The second proposed model is the layered generative error models. This has in particular three layers, whereby the first layer is comprised of one error-free burst state, and several error burst states assigned according to the maximum error cluster lengths. The second layer further divides the classes of error bursts based on their maximum gap lengths. The final layer constructs the error bursts.

- **Improving the deterministic process based generative model (DPBGM):**

  The DPBGM parameters are tuned to accommodate packet error profiles. Moreover, a modification to the DPBGMs has taken place in this thesis. A new packet-level generative model has been proposed to replace the problematic final design step (generation of error profiles) of the DPBGMs by using a HMM. The new packet-level generative model is called the Deterministic Process based Hidden Markov Model (DPB-HMM). The exact problem of the DPBGMs is that they do not create new error bursts on their own in the design step of generating error sequences; but only create the lengths of error bursts and error-free bursts. The DPBGMs subsequently retrieve error bursts of the same length directly from the reference error profiles. This behaviour restricts the capability of DPBGMs to generate new error sequences at different channel conditions when only a few reference error profiles are available. In addition, if only reference burst error statistics are given while reference error profiles are not available; the DPBGM is of no use. Therefore, a modification is needed to overcome the drawback of DPBGMs and to enhance the adaptability of the new model to generate error bursts and consequently error profiles at various channel conditions.

- **Designing adaptive generative models:**

  The adaptive generative model concept has been introduced and applied to widely used generative models, such as the Simplified Fritchman's Model (SFM), HMM, DPBGM, and DPB-HMM. Each of these generative models has been checked against the most effective featuring property. These properties have subsequently been studied to make them reproducible for various parameters of the wireless systems or different channel conditions rather than a fixed channel condition. There is no doubt that adaptive generative error models influence the speed of business as they significantly reduce the simulating time of a real wireless communication system when several error profiles at different channel conditions are needed.

- **Testing the adaptive generative models:**

  Adaptively generated uncoded error profiles at bit level have been fed into the digital channel in order to investigate their performance with a coding scheme, specifically the LTE turbo coding scheme. Moreover, adaptively generated error profiles at the packet level have replaced the entire physical layer in order to scrutinise their accuracy with higher layer protocols, particularly the LTE Hybrid Automatic Repeat reQuest (HARQ).

- **Predicting the performance of HARQ:**

  A statistical prediction model for a digital wireless channel with HARQ has been proposed to enable an understanding to be elicited regarding the burst error statistics of a digital wireless channel that includes HARQ from a digital wireless channel that does not include HARQ. Firstly we studied the statistical behaviour of digital wireless channels when they had or did not have HARQ. Then, we have found the statistical relationship between these digital channels when they did or did not have HARQ. This prediction model also has an impact on reducing the time of designing HARQ.

- **Studying and simulating a set of burst error statistics and widely known generative models:**

  Thorough investigations have been conducted to study the burst error statistics and then to simulate them. Moreover, studying and implementing several existing generative models have been required for understanding and comparison purposes. The aforementioned achievements in this thesis could not have been accomplished without these studies.

1.4 Publications

The work presented in this thesis has led to the following publications:

Journals

1. Y. He, **O. S. Salih**, C.-X. Wang, and D. Yuan, "Deterministic process based generative models for characterizing packet-level bursty error sequences," *Wireless Commun. and Mobile Computing*, 2013, DOI: 10.1002/wcm.2356.

2. **O. S. Salih**, C.-X. Wang, Y. He, R. Y. Mesleh, and D. Yuan, "A packet-level deterministic process based hidden Markov model for digital wireless channels," *IEEE Trans. Mobile Computing*, submitted for publication.

3. **O. S. Salih**, C.-X. Wang, B. Ai, and D. I. Laurenson,“Adaptive generative models for digital wireless channels,” *IEEE Trans. Wireless Commun.*, submitted for publication.

Conferences

1. **O. S. Salih**, C.-X. Wang, R. Mesleh, X. Ge, and D. Yuan,“Predicting burst error statistics of digital wireless systems with HARQ,” in *Proc. IEEE International Wireless Communications and Mobile Computing Conference, IWCMC 2013*, Cagliari, Sardinia, Italy, 1–5 Jul. 2013, pp. 276–281.

2. **O. S. Salih**, C.-X. Wang, D. I. Laurenson, and Y. He,“Hidden Markov models for packet-level errors in bursty digital wireless channels,” in *Proc. Loughborough Antennas and Propagation Conference, LAPC 2009*, Loughborough, UK, 16–17 Nov. 2009, pp. 385–388.

3. **O. S. Salih**, C.-X. Wang, and D. I. Laurenson,“Soft bit error modeling for discrete wireless channels,” in *Proc. IEEE International Wireless Communications and Mobile Computing Conference, IWCMC 2009*, Leipzig, Germany, 21–24 Jun. 2009, pp. 759–763.

4. **O. S. Salih**, C.-X. Wang, and D. I. Laurenson,“Three-layered hidden Markov models for binary digital wireless channels,” in *Proc. IEEE International Conference on Communications, ICC 2009*, Dresden, Germany, 14–18 Jun. 2009.

5. **O. S. Salih**, C.-X. Wang, and D. I. Laurenson,“Double embedded processes based hidden Markov models for binary digital wireless channels,” in *Proc. IEEE International Symposium on Wireless Communication Systems, ISWCS 2008*, Reykjavik, Iceland, 21-24 Oct. 2008, pp. 219-223.

1.5 Thesis Organisation

The remainder of this thesis is organised as follows:

Chapter 2 provides some essential background information related to the area of research presented in this thesis. Firstly, an introduction about digital wireless channels has been provided. Secondly, the error sequences (error profiles) which have been obtained from the digital wireless channels have been introduced. Analysing the error sequences and its statistics which is called "Burst error statistics" occupies part of this chapter. Finally, the description of error models has been provided with a general overview of generative error models categorised into five groups, in particular Markov models, hidden Markov models, stochastic context-free grammars, chaotic attractors generative models, and deterministic process based generative models.

Chapter 3 presents novel Markovian generative models that compromise very well between complexity and accuracy. The first model has two layers of processes, one for combining error-free bursts with error bursts which are divided into groups according to a specific criterion. The second process is dedicated to construct accurate error bursts to subsequently create accurate error profiles. This model has been scrutinised against hard and soft error profiles in addition to packet error profiles. Moreover, the criterion of grouping the error bursts in the first layer of this model has been varied to check the effect of the grouping criterion on the final results. The second model improves upon the aforementioned model by adding an extra layer of processes in order to further enhance the processing procedure.

Chapter 4 develops the DPBGM for the packet level and shows its performance at this level compared to the SFM and the BWHMM. Moreover, the existing DPBGM has been improved in order to advance its ability of generating the error bursts. All the aforementioned models are parametrised involving reference error profiles derived from EGPRS system. Moreover, the burst error statistics of the generated error profiles have been compared with those of the EGPRS system.

Chapter 5 introduces the concept of adaptive generative models and the motivation behind them. The most efficient features of widely-used generative error models, namely the SFM, BWHMM, DPBGM, and DPB-HMM have been studied. From the prevailing features trend and behaviour of few error profiles, new required error profiles have been created by interpolating the prevailing features trends for each generative error model. Surely, for parameterising the error models and studying the prevailing features, a few reference error profiles are required. Hence, an LTE system has been used to obtain uncoded reference error profiles for the aforementioned purposes as well as the comparison evaluation. The newly generated error profiles by adaptive means have replaced the digital wireless channel to investigate the LTE coding performance for different generative models.

Chapter 6 analyses the digital wireless channels when the HARQ is attached to the physical layer; whilst conversely, the digital wireless channel has been analysed when the HARQ is excluded. The statistical differences between the two digital channels have been noticed and studied. A prediction generative model has subsequently evolved which is able to predict the behaviour of HARQ while error profiles of the physical layer without HARQ are only available. Moreover, the entire physical layer is replaced by packet error profiles obtained from the developed adaptive procedures in order to scrutinise the generative models against the performance of HARQ. An LTE system has been utilised to attain reference error profiles for the above experiments.

Chapter 7 concludes the thesis and suggests some future research topics.

Chapter 2

Background

2.1 Digital Wireless Channels

Wireless communication systems have generally two types of channels, namely, analog (physical) and digital (time-discrete) [4–6]. For the analog channel, we are interested in the signal strength, signal-to-noise ratio (SNR) or carrier-to-interference ratio (CIR), and mobile speed, etc [7–10]. Digital wireless channels have time-discrete inputs and outputs. Thus, digital wireless channels comprise the transmission chain in a communication system including the transmitter, physical channel, and receiver. Hence, the physical channel is part of the digital wireless channel. Common parameters set for digital wireless channels are the number and distribution of error events within a sequence of bits or packets. A typical digital wireless channel is shown in Figure 2.1.

2.1.1 Error Sequences

An error sequence can be attained by comparing the digitised output of a digital channel with its digitised input sequence. Therefore, an error sequence is a series of consecutive digitised symbols. Error sequences can be considered at either bit level or packet level.

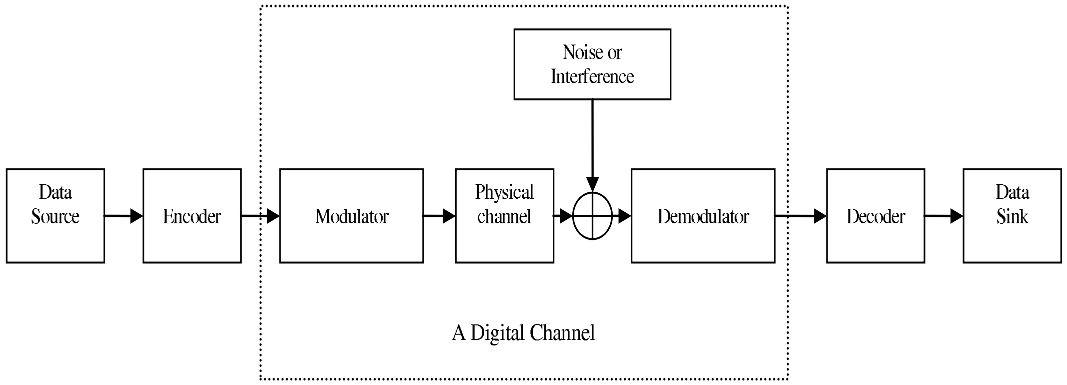

FIGURE 2.1: A typical digital communication channel.

...6 -3 7 -1 5 -5 3 6 -8 -4 -2 6 4 5 7 -2 1 7 4 1 -8 -2 -3 -5 -1 -1 2 5 3 6 1 7 -1 -4 -5 4...

FIGURE 2.2: Soft error sequence example ($\eta = 4, M = 4$).

Bit error sequences can be either soft or hard, depending on the type of the decision at the output of a digital channel.

A soft error sequence consists of real integers as FIGURE 2.2 points up. The values of the input sequence are in fact binary integers belong to -1, 1. However, the output sequence consists of integer numbers ranging from -2^{M-1} to $2^{M-1} - 1$, where M is a positive integer [6]. Mathematically, a soft error sequence is obtained by multiplying the input sequence with output sequence. The resulting negative integer indicates an error bit, whereas a non-negative integer implies a correctly received bit.

A hard error sequence, on the other hand, is a binary sequence. This means that the values of the elements of the hard error sequence belong to 0, 1 as FIGURE 2.3 illustrates. This means that the input and output to the underlying digital channel belong to 0, 1 as well. A binary error sequence can be mathematically worked out by carrying out modulo 2 addition between an input sequence and output sequence. The “0”s and “1”s correspond to correct bits and error bits, respectively. Please notice that a hard error sequences is a quantised version of soft error sequence, i.e.,

...0 1 0 1 0 1 0 0 1 1 1 1 0 0 0 0 1 0 0 0 0 1 1 1 1 1 1 1 1 0 0 0 0 0 0 1 1 1 1 0...

FIGURE 2.3: Hard error sequence example ($\eta = 4$).

non-negative integers can be quantised to ‘0’, and negative integers can be quantised to ‘1’.

In order to understand the error sequences and their burst error statistics, some terms related to the error sequences must be introduced.

2.1.1.1 Error Sequence Terms

In this section the terms of error sequence, which are demonstrated in Figure 2.4, and the burst error statistics which are used in the simulations are defined.

A *gap* (G) is a series of successive zeros (non-negative integers) between two ones (negative integers). The gap length is the number of zeros (non-negative integers). An *error cluster* (EC) is a series of successive ones (negative integers). It has a length equal to the number of ones (negative integers). An *error-free burst* (EFB) is a sequence of successive zeros (non-negative integers) that have at least η symbols length. It is different from the gap in the length restriction. Also, it is not necessary for an error-free burst to be located between two ones (negative integers). An *error burst* (EB) is a sequence of zeros (non-negative integers) and ones (negative integers).

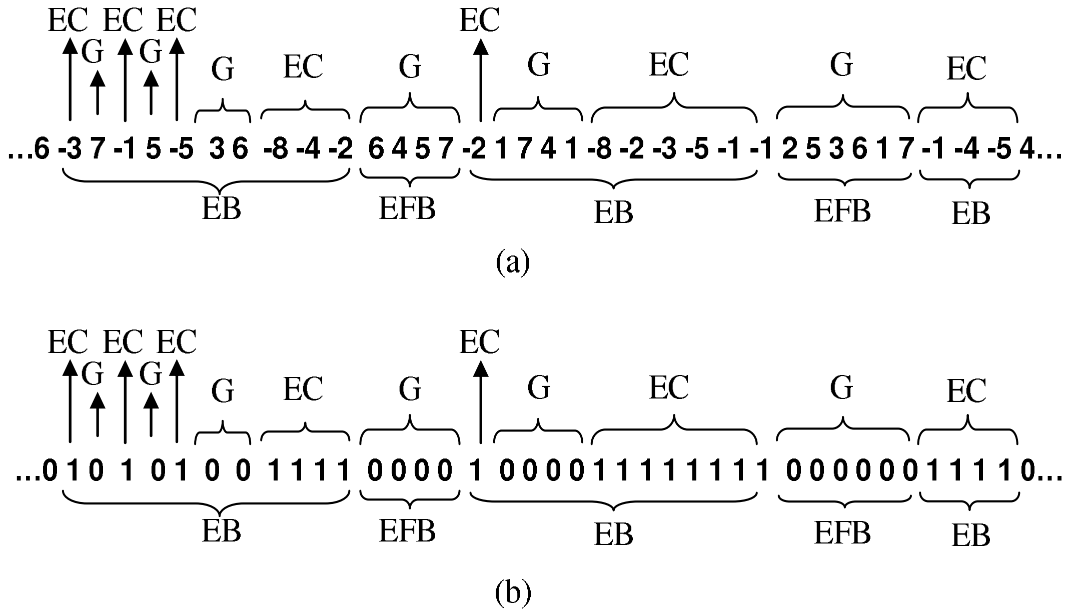

Figure 2.4: (a) Soft error sequence example ($\eta = 4, M = 4$). (b) Hard error sequence example ($\eta = 4$).

It starts and ends with ones (negative integers), and separated from other error bursts by error-free bursts.

The error burst lengths and error-free burst lengths of a reference error sequence can be recorded as vectors called $\mathbf{EB}_{rec}$ and $\mathbf{EFB}_{rec}$, respectively. We 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 $m_{\bar{B}1} \;\leq\; m_{\bar{e}} \;\leq\; m_{\bar{B}1}$.

The following quantities are related to the reference error sequences:

1) N_t: the total length of the reference error sequence.

2) $\mathcal{N}_{EB}$: the total number of error bursts, which equals the number of entries in $\mathbf{EB}_{rec}$.

3) $\mathcal{N}_{EFB}$: the total number of error-free bursts, which equals the number of entries in $\mathbf{EFB}_{rec}$.

4) $\mathcal{R}_B$: the ratio of the mean value M_{EB} of error burst lengths to the mean value M_{EFB} error-free burst lengths, i.e., $\mathcal{R}_B = M_{EB}/M_{EFB}$.

To study the error sequences comprehensively, error models need to be introduced.

2.2 Error Models

Errors which occur in digital wireless channels are not random, but appear in bursts or clusters, such that long error-free bursts are followed by error bursts or clusters. Models that can characterise the statistical behaviour of bursty error sequences in digital wireless channels are basically called error models. Error models also exploit the characterised statistics in order to produce required error sequences.

In general, error models contain two categories: descriptive (reference) and generative (simulation) models [11, 12]. Descriptive models identify the statistical behaviour of reference error sequences, which can be obtained from real digital wireless channels or from computer simulations implementing the whole communication link. Generative models, on the other hand, describe a procedure or methodology that can generate error sequences which have approximately similar error burst statistics to those of the reference error sequences.

Error models have two main advantages. Firstly, they are significant in the design of error control schemes and higher layer protocols, where the burst error statistics are beneficial. Secondly, the underlying digital wireless channels can be replaced by generated error sequences. Consequently, the computational load and simulation time can be greatly reduced, especially when performing evaluation simulations on different error schemes [13–16] and communication protocols [17–20]. In other words, there is no need to transmit the data again when a new test for error schemes or protocols is established. The replaced error sequences can perform instead of the digital wireless channel which is the core part of the wireless communication system.

2.2.1 Burst Error Statistics

Burst error statistics are the crucial metrics for criticising any generative error model and are considered the basis for the comparison between different generative models. Some of them are essential for the design and performance evaluations of parts of the digital wireless channel. Moreover, the majority of them demonstrate the structural nature of various error profiles.

Since error sequences could be hard bits, soft bits, or packets, the burst error statistics can be divided into two categories: binary burst error statistics and soft burst error statistics.

2.2.1.1 Binary Burst Error Statistics

Binary burst error statistics cover the hard bit error sequences as well as the packet error sequences because both types of error sequences are represented by “0”s and “1”s only.

To illustrate the binary burst error statistics we define them and show their figures at specific conditions in the following. We obtain reference error sequences from an uncoded Enhanced General Packet Radio Service (EGPRS) system with ideal frequency hopping (IFH). This system was built in the baseband algorithms and standardisation laboratory, Siemens AG-Mobile phones, Munich, Germany, within the framework of the 3GPP GERAN system concept R&D project. In our simulations, the underlying physical channel is tailored to a typical urban (TU) environment, and the mobile speed is 3 km/h. The CIR values vary at 5, 8, and 15 dB. The figures are for hard bit error sequences. The binary burst error statistics are:

1) $G(m_g)$: the gap distribution (GD), which is defined as the cumulative distribution function (CDF) of gap lengths m_g [21].

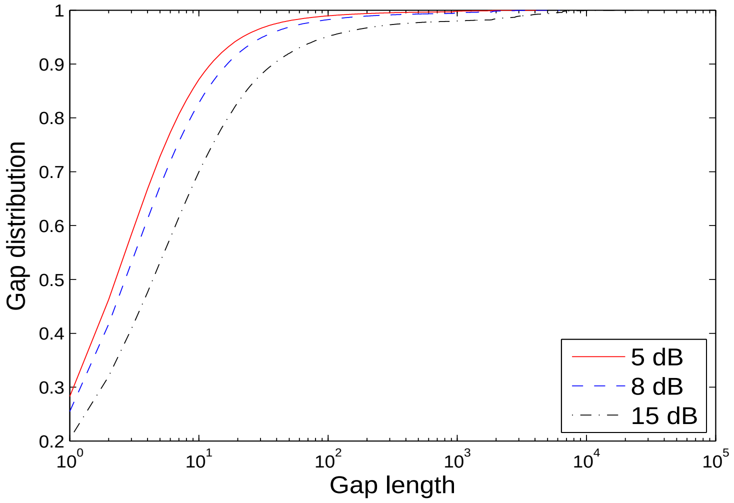

FIGURE 2.5: GDs of the descriptive model obtained from the EGPRS system at different CIRs.

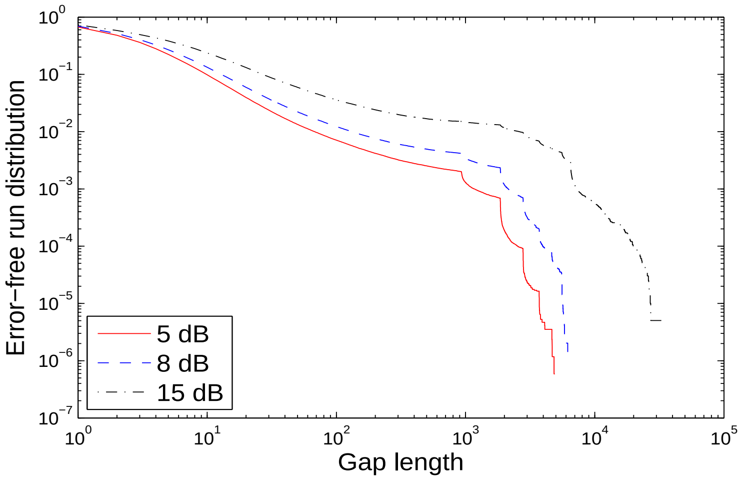

FIGURE 2.6: EFRDs of the descriptive model obtained from the EGPRS system at different CIRs

2) $P(0^{m_0}|1)$: the error-free run distribution (EFRD), which is the probability that an error ("1") is followed by at least m_0 error-free digits ("0"s) [22]. The EFRD can be calculated from the GD [21]. Obviously, $P(0^{m_0}|1)$ is a monotonically decreasing function of m_0 such that $P(0^0|1) = 1$ and $P(0^{m_0}|1) \to 0$ as $m_0 \to \infty$.

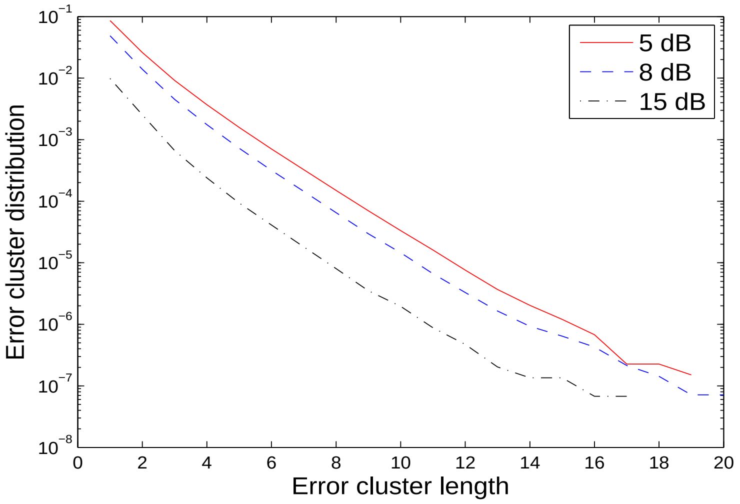

FIGURE 2.7: ECDs of the descriptive model obtained from the EGPRS system at different CIRs.

3) $P(1^{m_c}|0)$: the error cluster distribution (ECD), which is the probability that a correct digit is followed by m_c or more consecutive digits in error [22].

4) $P_{EB}(m_e)$: the error burst distribution (EBD), which is the CDF of error burst lengths m_e.

5) $P_{EFB}(m_{\bar{e}})$: the error-free burst distribution (EFBD), which is the CDF of error-free burst lengths $m_{\bar{e}}$.

6) $P(m,n)$: the block error probability distribution (BEPD), which is the probability that at least m out of n digits are erroneous. This statistic is important for determining the performance of Automatic Repeat Request (ARQ) protocols [23].

7) $Q(l,n)$: the block burst probability distribution (BBPD), which is the probability of an error burst of length l occurring in a block of length n. For only this statistic, the length of a burst in a block of n digits: is the number of zeros and ones between the first error to the last error in the block (both errors included) irrespective of the nature of the digits in between [12].

8) $Cov(l)$: the normalised covariance function (NCF) [24].

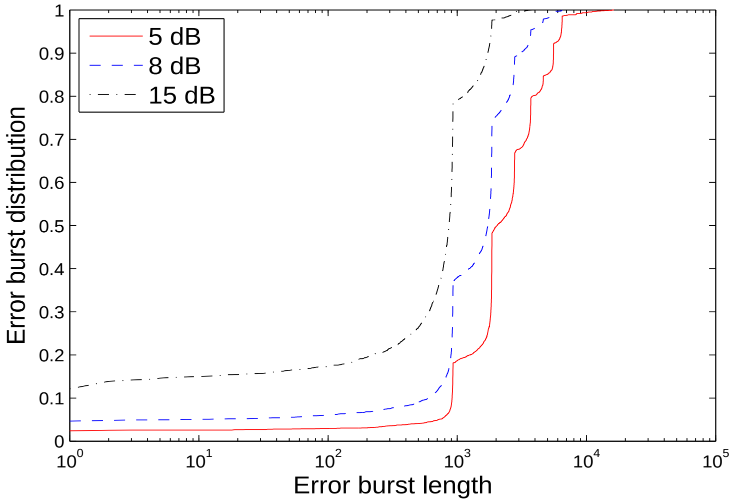

FIGURE 2.8: EBDs of the descriptive model obtained from the EGPRS system at different CIRs.

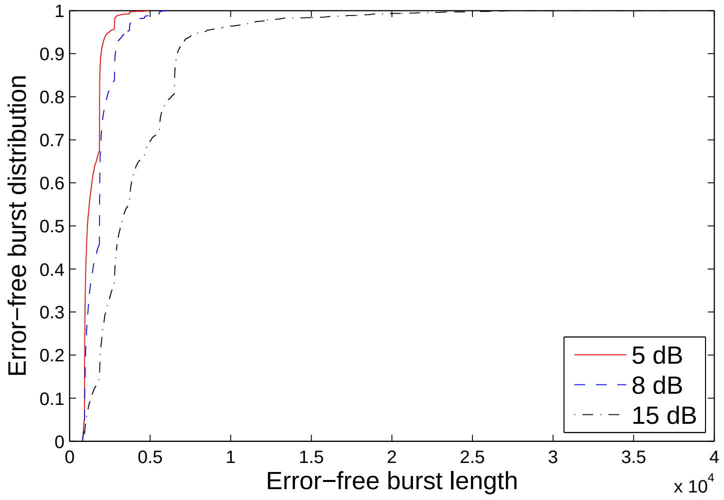

FIGURE 2.9: EFBDs of the descriptive model obtained from the EGPRS system at different CIRs.

9) $\rho(\Delta k)$: the bit (or packet) error correlation function (BECF or PECF), which is the conditional probability that the Δkth bit (or packet) following an error bit (or packet) is also in error. This statistic represents the burstiness of the channel [11, 12].

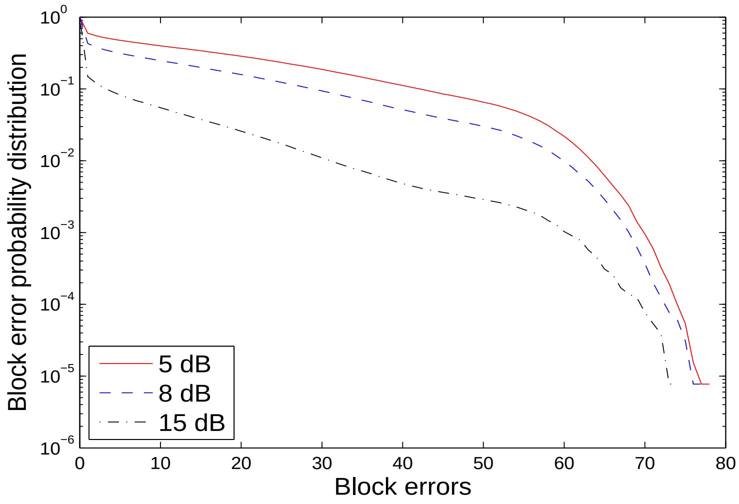

FIGURE 2.10: BEPDs of the descriptive model obtained from the EGPRS system at different CIRs ($n = 116$).

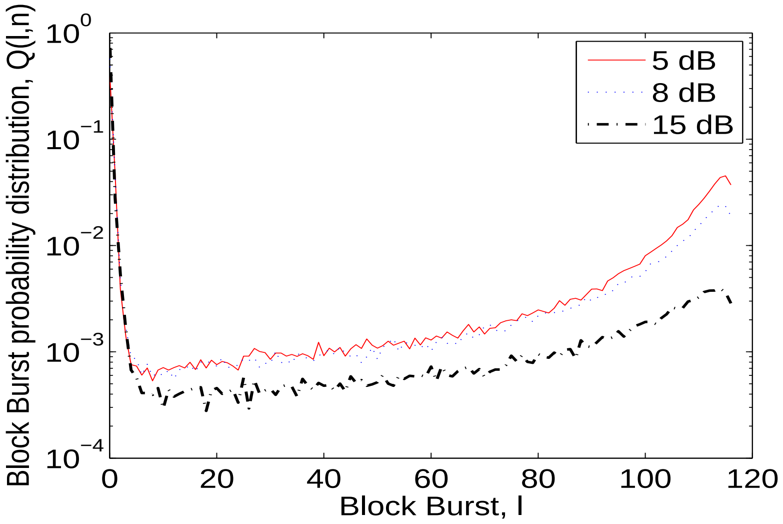

FIGURE 2.11: BBPDs of the descriptive model obtained from the EGPRS system at different CIRs.

10) *GCF*: the gap correlation function, which is the conditional probability that the Δrth gap following a short (long) gap is also short (long) [12].

11) *MGD*: the multigap distribution, which is the CDF of r consecutive gaps, considered as a single parameter, which are separated by one or more consecutive errors [12]. The gap here is different from the one adopted before, it is defined

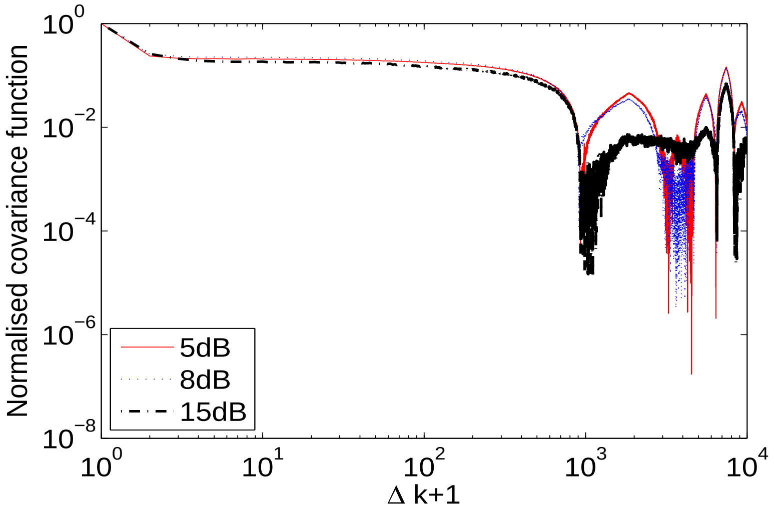

FIGURE 2.12: NCFs of the descriptive model obtained from the EGPRS system at different CIRs.

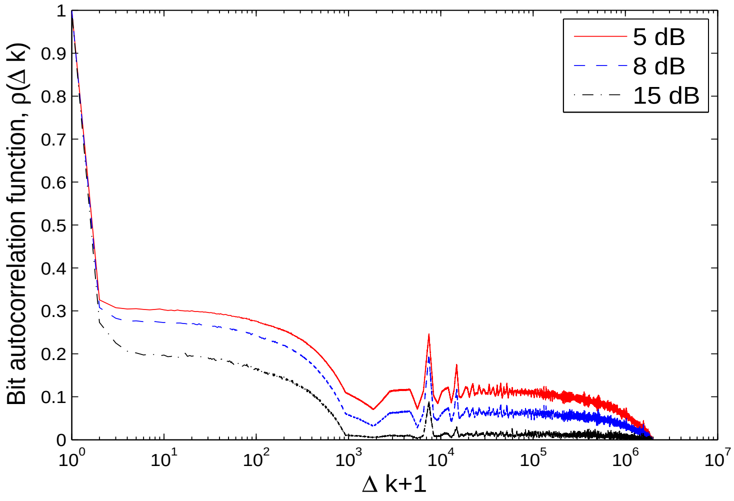

Figure 2.13: BECFs of the descriptive model obtained from the EGPRS system at different CIRs.

here as: a string of consecutive zeros between two errors and having a length equal to one plus the number of zeros between the two errors. It can be seen that the minimum value for a gap length is one, occurring in case of consecutive errors.

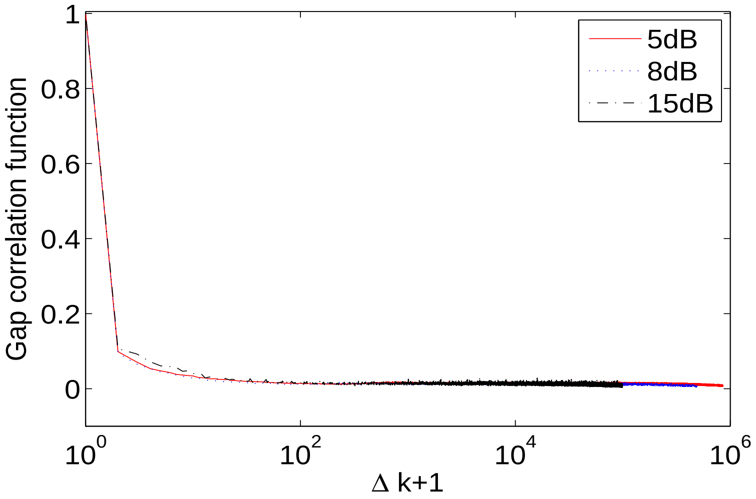

Figure 2.14: GCFs of the descriptive model obtained from the EGPRS system at different CIRs.

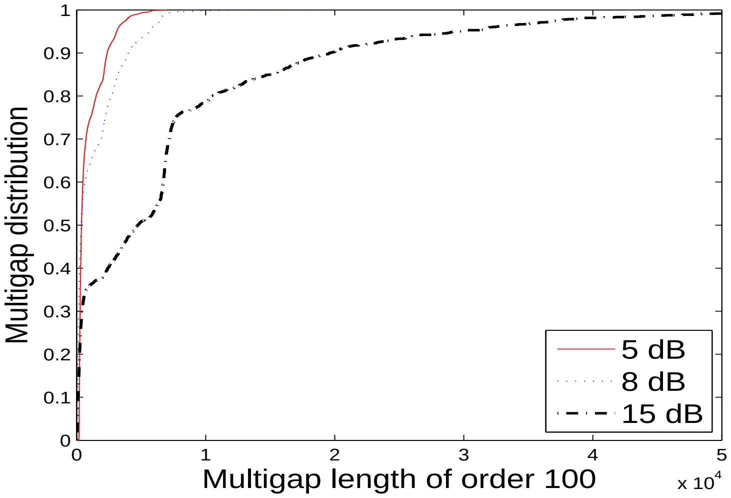

FIGURE 2.15: MGDs of order 100 for the descriptive model obtained from the EGPRS system at different CIRs.

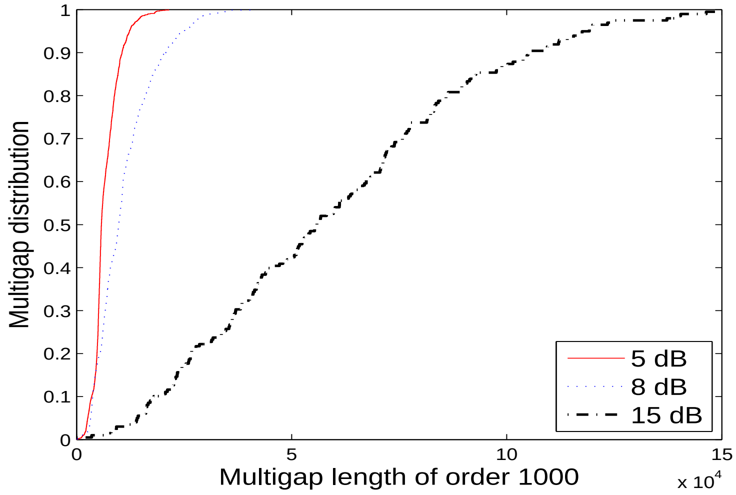

FIGURE 2.16: MGDs of order 1000 for the descriptive model obtained from the EGPRS system at different CIRs.

12) $K(r)$: the variation coefficients [12], which compare, for any channel C and its corresponding binary symmetric channel (BSC), with the same BER, the spreads of the multigap-length distributions around their common mean value r/p, where p is the average probability of error.

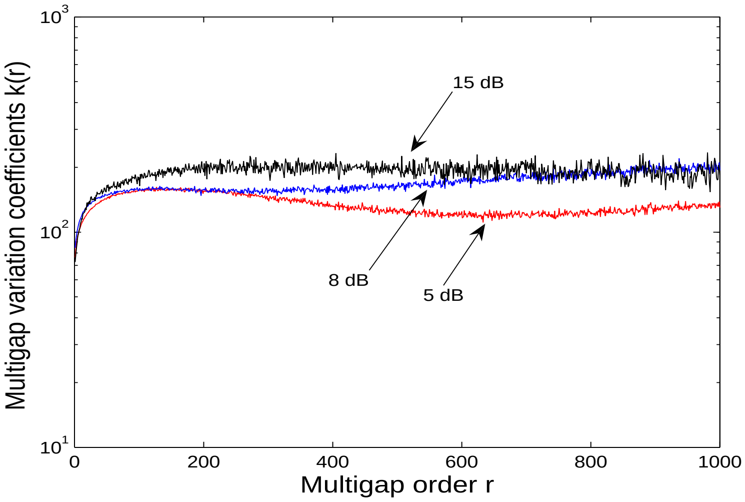

Figure 2.17: The variation coefficients of the descriptive model obtained from the EGPRS system at different CIRs.

2.2.1.2 Soft Burst Error Statistics

Soft burst error statistics analyse the soft error sequences. Some of these statistics are defined as follows.

1) $P(m_+)$: the soft error-free run distribution (SEFRD), which is the probability that a negative integer is followed by m_+ or more non-negative integers.

2) $P(m_-)$: the soft error-cluster distribution (SECD), which is the probability that a non-negative integer is followed by at least m_- successive negative integer.

3) $G(m_g)$: the soft gap distribution (SGD), which is the cumulative distribution function (CDF) of gap lengths m_g.

4) $P(m, n)$: the soft block error probability distribution (SBEPD), which is the probability that a block of n integers contains at least m negative integers.

5) $\rho(\Delta k)$: the soft bit-error correlation function (SBECF), which is the conditional probability that the Δkth integer following a negative integer is also a negative integer.

6) $P(e_j)$: the soft decision-symbol distribution (SDSD), which is the CDF of soft decision symbols $e_j \in [-2^{M-1}, 2^{M-1} - 1]$.

2.3 Overview of Generative Models

The literature is rich with many generative models that have been proposed during the last century in order to create accurate and efficient error sequences to achieve the purposes from there innovation. Generative models can in general be classified into five categories as briefed in the following subsections.

2.3.1 Markov Models

Markov models (MMs) are based on Markov chains which follow the well known Markov property. The first basic generative model of this category is called Gilbert's model [25], which is composed of two states: good (G) and bad (B) as shown in Figure 2.18. For the G state, the produced error digit is always 0, whereas for the B state the output digit is 0 or 1. The transitions between the states are carried out according to transition probabilities in order to construct an error sequence. The probabilities for persisting in the same state are greater than the other probabilities which are for transferring to the other state. Elliot [26] suggested generating errors in both of the two-states of the Gilbert's model, but the generated errors in the G state have low probability and hence quantity. The two-state models have received many applications due to their simplicity. However, these models are disappointing in terms of accuracy as they failed to generate error sequences with burst error statistics closer to those of the original error sequences. The two-state models have a simple structure and basic parameters. They also have renewal natures which limit their accuracy. Moreover, they are parametrised based on the assumption of having a geometric distribution for the run lengths of both states.

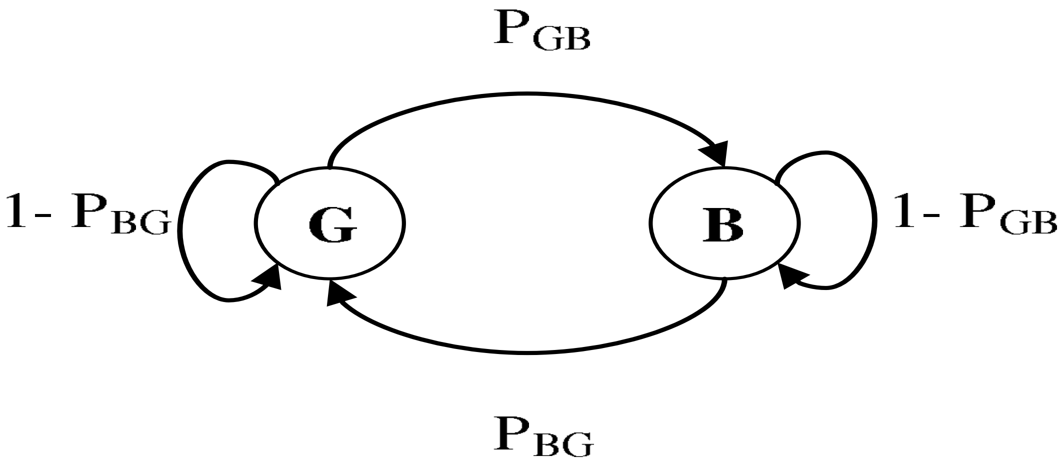

FIGURE 2.18: Gilbert-Elliot (two state) generative model.

Thereafter, several modifications to the simple MMs have been proposed in order to improve their performance. These modifications include increasing the number of states[27–36] or imposing certain conditions on the transitions [37–43]. Berkovits et al. [27] proposed a model that consists of three states. Two states are bad and the third is good. The error gap distribution for this model is the sum of three exponential terms. In fact, Berkovits' model failed to obtain satisfactory results. McCullough[28] proposed to further increase the number of states. In his model, errors can occur in all the states with different error rates, and the transitions between the states are allowed only immediately following an error. Trafton et al. [37] suggested a model with similar idea to McCullough's, but his model has only two states. For Trafton's model, the transition probabilities are functions of sojourn times in each state.

A well-known Markovian model is the Fritchman's partitioned finite-state model[22]. It is composed of N-states divided into two groups of k error states and $N - k$ error-free states. Fritchman's model has a simplified and widely used version called the Simplified Fritchman's model (SFM).

Another Markovian model which is based on infinite states was proposed by Adoul et al. [38]. It consists of two coupled renewal infinite state Markov chains. For this model, the transitions between the chains can happen only after an error occurs. The transitions are also conditioned on the length of the previous gap. A higher order MM which is consisting of N-states was suggested by Blank et al. [39]. Only one transition step is allowed in this model. An N-state model with transition probabilities based on a conditional gap distribution was proposed by Chein et al. [40]. An extension to this model with additional memory incorporated for short gaps was recommended by Varshney et al. [43]. A major drawback of the infinite states Markov models is their extreme complexity. It is also recognised that the conditional and unconditional gap distributions for the infinite states Markov models are in the form of sums of exponentials, which are similar to the gap distributions of Fritchman's model. This is a reason which could have given Fritchman's model a pioneer over all the other MMs. Fritchman's model also gives a good compromise between simplicity and accuracy.

2.3.1.1 Simplified Fritchman's Models (SFMs)

As previously mentioned, a SFM (see Figure 2.19) consists of N-states; one error state and $N-1$ error-free states that characterise the error cluster distribution and error-free distribution, respectively.

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 SFMs were applied to many channels, such as HF [21], VHF [23], UHF [44], and indoor channels [24, 45]. It is found that, SFMs are not applicable to slow fading scenarios, whereas they are more applicable to scenarios of intermediate to fast fading [23].

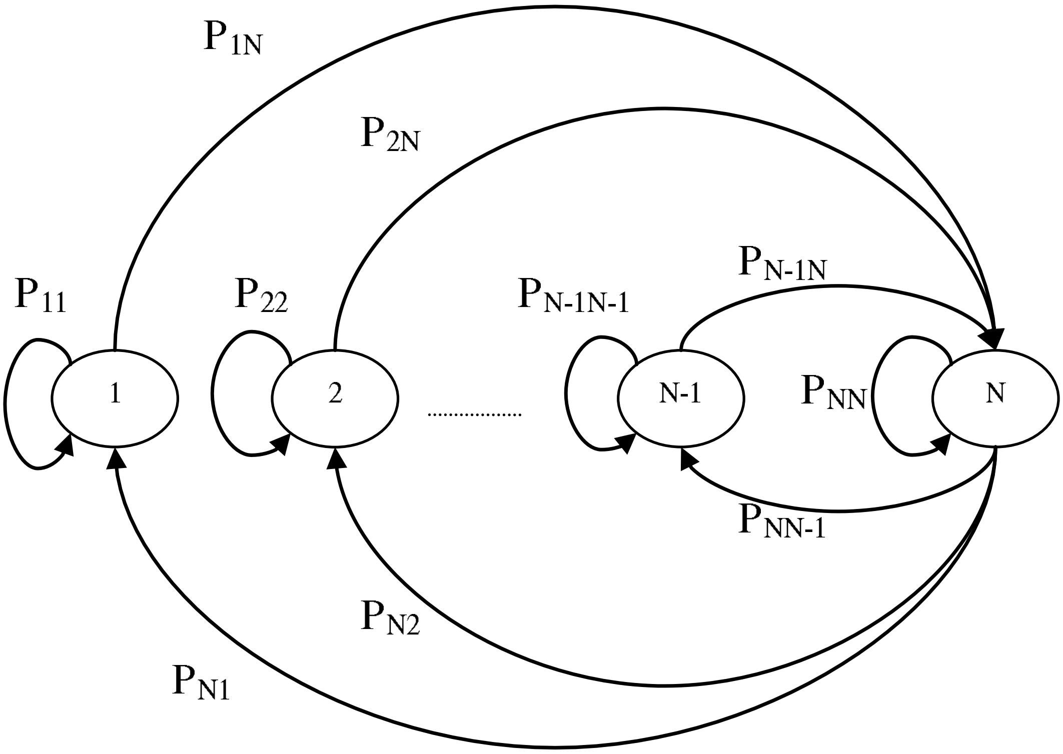

FIGURE 2.19: States diagram of the SFM.

2.3.2 Hidden Markov Models

Hidden Markov models (HMMs) [24, 45–56] employ the idea of Markov models, but with using two stochastic processes. One of the stochastic processes is not observable, but can only be observed by another stochastic process which produces a sequence of observations. The HMMs are most likely used for burst error characterisation in indoor radio channels. One of the methods used to implement this class is the one mentioned in [45] and is called Baum-Welch based HMMs (BWHMM). For these models, the error bursts in the reference error sequence are extracted and numbered. They are then divided into blocks of length L bits. Each block is marked by the number of errors it contains (compacted format), as in FIGURE 2.20. The largest compacted number in each error burst called the Peak Number of Errors (PNE).

After that, the error bursts are classified into j disjoint classes (bursty classes) using the PNE as a criterion in such a way that $\xi(j-1)+1 \;\geq\; \text{PNE} \;\geq\; \xi j$, where ξ is a positive integer number. Then, the compacted blocks of each class are used to

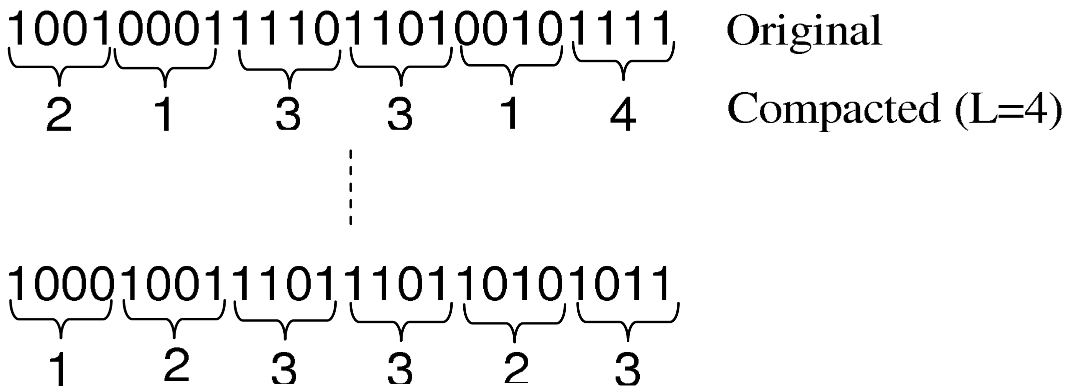

Figure 2.20: Error bursts in compacted format.

train hidden Markov submodels using the Baum-Welch algorithm [47, 57, 58]. Each submodel contains one class of error bursts. Consequently, many error bursts can be generated from the submodels after the training. From the mathematical point of view, the BWHMMs have the following elements:

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 [45]. Given a set of observation sequences representing the compacted error burst $\mathbf{O}^k = \{O_1^k, O_2^k, \cdots, O_{D_k}^k\}$, $k = 1, \ \cdots \ , \ K$ (K is the number of error bursts in each class), the BW algorithm is utilised to maximise the probability $\Gamma = \prod_{k=1}^{K} P(\lambda|\mathbf{O}^k)$.

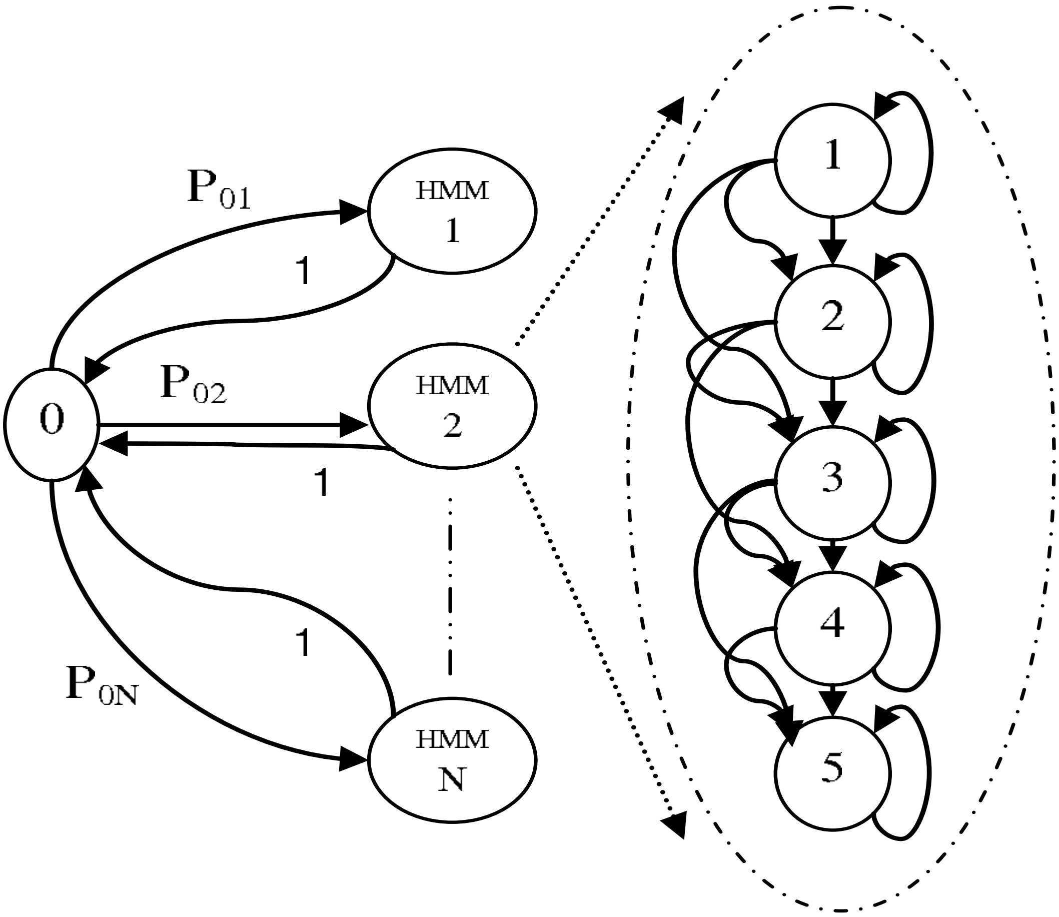

FIGURE 2.21: The BWHMM states concatenation.

Finally, an error-free bursts concatenation to the BWHMM submodels is executed. FIGURE 2.21 shows the concatenation implementation of the error-free burst state with the error bursts submodels. Each submodel represents one class and has one state. The transitions from the error-free state to the other states generate error bursts with variable structures according to the states of transition. Whereas, the transitions from the error burst states to the error-free state generate error-free bursts with different lengths.

In fact, the Baum-Welch algorithm is robust since it always converges, but it cannot guarantee that the convergence point is always a global maximum. Thus, its final parameters may not be the optimal ones. Another drawback is that the BWHMMs consist of a large number of states. This in turn increases the complexity of the model. Furthermore, the BWHMMs are best for characterising the error bursts of indoor channels, where the error profiles have long error bursts and the errors' densities in the error bursts are following the bell-shape.

2.3.3 Stochastic Context-Free Grammars

Stochastic Context-Free Grammars (SCFGs) are set of rewriting rules called productions. Each production is augmented with a probability. The probability of a derivation (parse) is then the product of the probabilities of the productions used in that derivation [59].

SCFGs can be utilised in order to generate error sequences [24]. Similar to the HMMs, the SCFG models are only able to portray the statistics of error burst profiles of indoor environments, where the error-density behaviour is bell-shaped. This limitation reduces the applicability of these models. The HMMs were afterwards modified based on the idea of context-free grammars. Nonetheless, the advantage of the SCFGs over the HMMs is that they are much more powerful in modelling palindromes. Therefore, the SCFGs make the modelling of the bell-shaped error density inside an error burst to occur in a natural and easy way. They can generate the ascending and descending parts inside an error burst at the same time.

FIGURE 2.22 shows an SCFG and modified HMM. The starting point in the processing of both the SCFG models and modified HMMs is to extract the error bursts from the original error sequence. Each error burst is partitioned into blocks of length L bits. A compact format representation can then be obtained by counting the number of errors inside each block of L. Consequently, the bursts are grouped into classes. FIGURE 2.22 (a) illustrates the SCFGs. They are made up of terminal and nonterminal symbols. The terminal symbols contain all the symbols that can appear in the reference sequence. Whereas, the nonterminal symbols are those which appear in the process of generating error sequences and are always replaced by a string of terminal and nonterminal symbols. In the SCFG models, a nonterminal symbol S_0 is associated with an error-free interval. A transition from S_0 to a nonterminal symbol S_1 identifies the beginning of an error burst, for which the ascending and descending

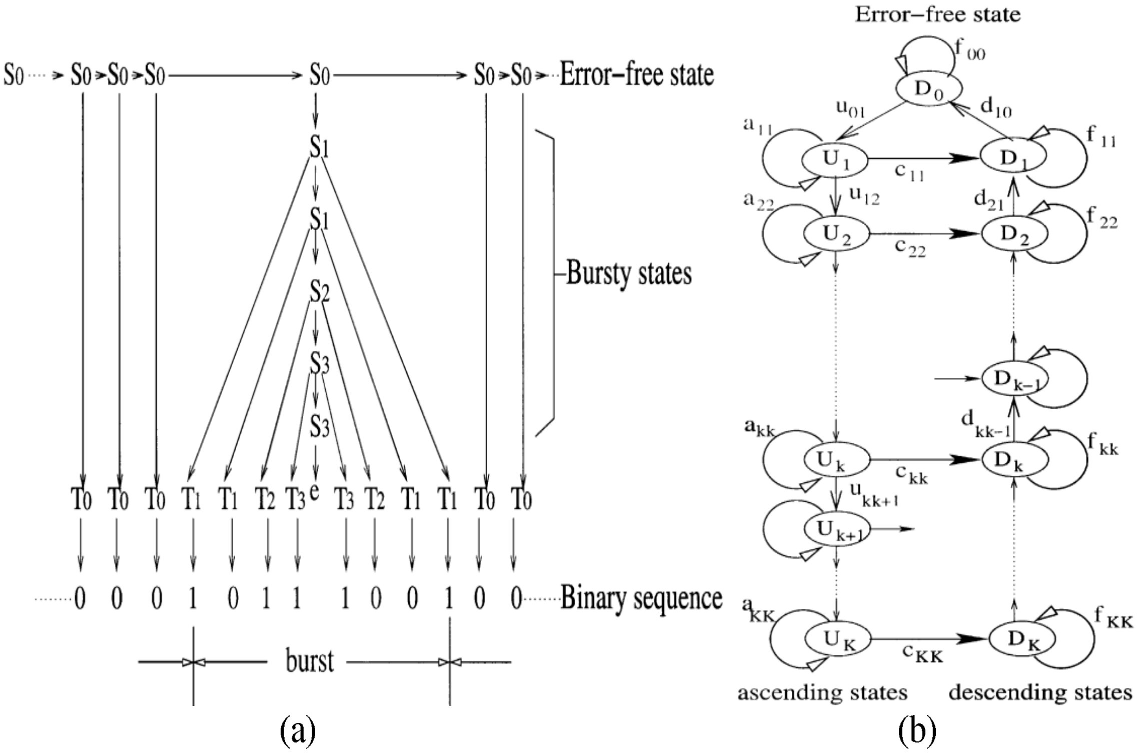

FIGURE 2.22: (a) Error sequence generation for the SCFG models. (b) State diagram for the modified HMMs.

parts of an error burst can be generated at the same time. Each error burst class is allocated a nonterminal symbol S_k, which has a probability to terminate the error burst generation. Then, the state S_0 resumes and the process starts again until the desired error sequence length is achieved. Here, k is a positive integer.

The modified HMMs [24] are demonstrated in FIGURE 2.22(b). The state D_0 generates error-free intervals. It has a small transition probability to state U_1 which begins an error burst generation. For each error burst class, there are U_k and D_k states, which generate the ascending and descending parts of an error burst, respectively. The state U_k has a transition probability to U_{k+1} to continue forming the ascending part or D_k to initialise the descending part, whereas D_k has a transition probability to D_{k-1} to continue forming the descending part. Once the model reaches D_1, there is a probability to stay at D_1 for sometime or to terminate the error burst generation, then the error-free state D_0 resumes. The process continues till an error sequence with a specified length is obtained. By the way, it is realised that the modified HMMs outperform the SCFG models regarding the burst error statistics results [24]. Therefore, we will not consider this class of models in our future analysis and comparison.

2.3.4 Chaotic Attractors Generative Models

The chaotic attractors generative models (CAGMs) are based on chaotic systems [60–64]. In fact, solving a differential equation that analyses a physical system in the time domain (continuous or discrete) gives a solution which can be stable, periodic, quasi-periodic, or chaotic. The chaotic solution should be treated by methods of linearisation to get more acceptable resolutions.

A chaotic attractor is considered as a subclass of chaotic systems. It can be considered as an intersection of a mathematical chaos and topological geometry. The aim of this approach is to exploit the properties of a chaotic system to generate a time series of arbitrary events with given statistics. The attractor of interest is the Lorenz attractor which is shown in FIGURE 2.23. It is worth mentioning that, the chaotic system exhibits a random behaviour in spite of its deterministic nature.

To obtain meaningful numbers from a chaotic attractor in the target environment, sampling should be done for the trajectories. Consequently, projections of the sampled points on the system axes should be calculated. In this case, the projections resemble the bell-shaped Gaussian noise distribution, and then the probability of the error can be determined by the tails of the distribution. A PDF of the coordinate variable can be estimated in the tail region. After that, the minimum distance between the estimated error gap distribution and the one of the reference can be calculated by means of "Nelder and Mead" algorithm or other optimisation procedures. Another way to follow is to use several attractors in which the sampling can be performed to all of them, but with different steps. The obtained coordinates are added up using a cost function that weights the mix. A PDF of the gap lengths can then be calculated. After that, a shaping function is considered to match the error gap distribution to that of the reference time series. By sampling the attractors again, the model generates

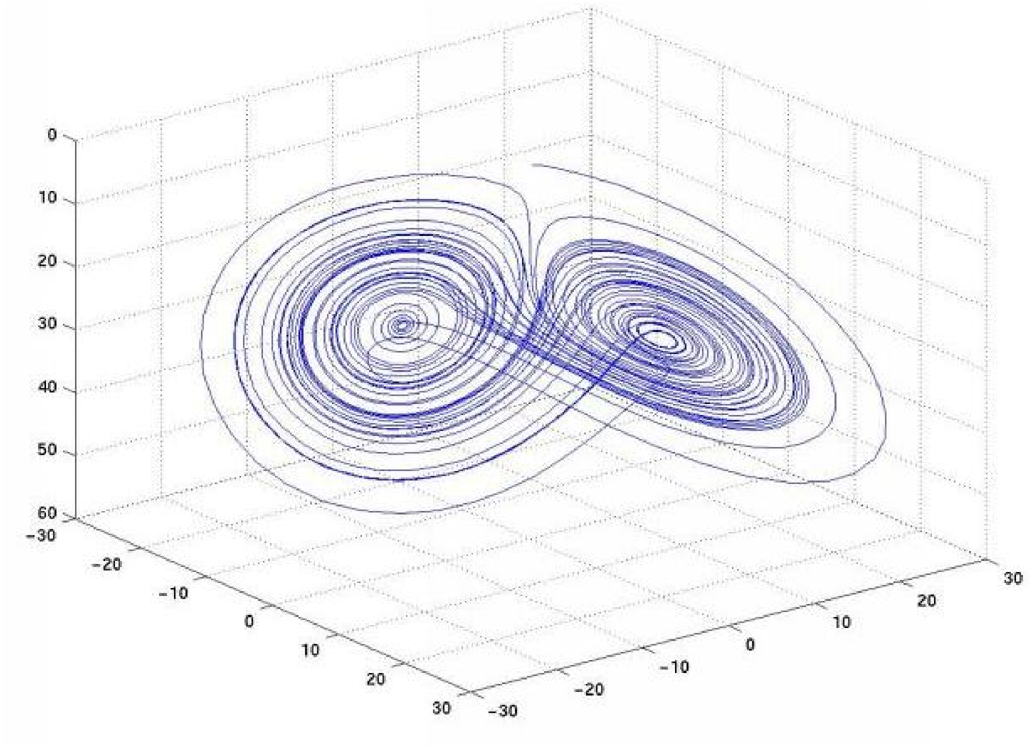

FIGURE 2.23: Lorenz attractor.

a new realisation of the error process, which is different from the reference one but exhibiting, generally speaking, similar statistics. In reality, the discrepancy between the reference burst error statistics and those obtained from the attractor is quite high, especially for the most important burst error statistics such as the bit error correlation function. This method in general depicts the following equation:

$$l = S[f(\sum_i (k_i u_i))], \tag{2.1}$$

where $f(\cdot)$ is an exponential or a polynomial function for the projections of the vectors joining the current point of the trajectory and the coordinate origin, $S[\cdot]$ is a shaping function to match the gap lengths probability distribution to that obtained from the reference error sequences, k is a suitable weight, u is the x, y, or z coordinates and i denotes the number of attractors.

Despite that this model has a new idea rather than the well-known Markov models, it is considered very complex and it takes long time to execute. Furthermore, it lacks the accuracy in describing important burst error statistics, such as the bit error correlation function as mentioned before. Consequently, we will not consider this class of models any further.

2.3.5 Deterministic Process Based Generative Models

The final category, we consider in this chapter, is called the deterministic process based generative models (DPBGMs) [65–70]. These models are related to the fading processes, which can be represented by deterministic processes [7, 71] that are based on the rule of sum of sinusoids [72–76]. The deterministic process is called as such because all of its parameters are held constant during the simulation. In fact, some statistics of bursty error sequences, namely the error burst and error-free burst distributions, can be approximated from the second order statistics of fading envelope processes, more precisely the level crossing rate (LCR) and the average duration of fades (ADF). The error burst and error-free burst lengths are associated with the fading and inter-fading intervals. Hence, fading processes can be used to generate error sequences.

To build a DPBGM, an underlying reference transmission system is replaced by a properly parametrised and sampled deterministic process followed by a threshold detector and two parallel mappers to fit the obtained length distributions of the error and error-free bursts to the desired statistics of the descriptive models. Figure 2.24 demonstrates the DPBGM.

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 created. After that, a

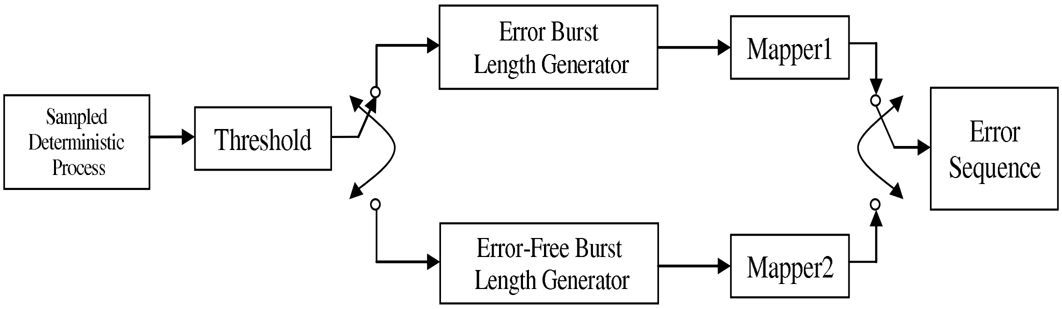
Sampled Deterministic Process
Threshold
Error Burst Length Generator
Error-Free Burst Length Generator
Mapper1
Mapper2
Error Sequence

FIGURE 2.24: The DPBGM implementation.

mapping process is forced to adjust the generated error burst and error-free burst lengths to those of the reference error sequence. At the end, error sequences can be obtained by consecutive combining of the generated error bursts with the generated error-free bursts.

The deterministic process can be represented by:

$$\tilde{\zeta}(t) = |\tilde{\mu}_1(t) + j\tilde{\mu}_2(t)| \tag{2.2}$$

where

$$\tilde{\mu}_i(t) = \sum_{n=1}^{N_i} c_{i,n} \cos(2\pi f_{i,n} t + \theta_{i,n}) \ , \quad i = 1, 2 \tag{2.3}$$

here N_i is the number of the required sinusoids and $\theta_{i,n}$ are phases uniformly distributed over $(0, 2\pi]$. The discrete frequencies $f_{i,n} = f_{max} \sin\left[\frac{\pi}{2N_i}(n - \frac{1}{2})\right]$ where f_{max} represents the maximum Doppler frequency. The gains $c_{i,n} = \sigma_0\sqrt{\frac{2}{N_i}}$ where σ_0 is the square root of the mean power of $\tilde{\mu}_i(t)$.

Some second order statistics of the sampled deterministic process, such as the LCR, the ADF, and the average duration of the inter-fades (ADIF) can be described using the parameters in $\mathbf{\Psi} = (N_1, N_2, r_{th}, \sigma_0, f_{max}, T_A)$, where r_{th} is the threshold. The values of N_1 and $N_2 \geq 7$ to compare the deterministic process statistics with those of the Rayleigh process [70]. Consequently, the parameter σ_0 can be calculated through:

$$\sigma_0 = \frac{r_{th}}{\sqrt{2\ln(1+\mathcal{R}_B)}} \, . \tag{2.4}$$

The sampling interval T_A is defined as:

$$T_A \approx \frac{4\sigma_0[\exp(\frac{r_{th}^2}{2\sigma_0^2})-1]}{\sqrt{5}\pi r_{th} f_{max}} \sqrt{-1+\sqrt{1+10q_s/3}} \, , \tag{2.5}$$

where q_s is a very small quantity determining the maximum measurement error of the LCR and

$$f_{max} = \frac{\mathcal{N}_{EB}(1+\mathcal{R}_B)}{T_t\sqrt{2\pi\ln(1+\mathcal{R}_B)}} \, , \tag{2.6}$$

where T_t is the total transmission time of the reference system.

In the aforementioned procedure, a careful selection should be taken into account when deciding the value of the threshold and the parameters of the sampled deterministic process. The threshold is chosen such that it is much less than 1. The other parameters of the deterministic process were obtained by adapting the ratio of ADF and AIDF at a chosen threshold to $\mathcal{R}_B$. Moreover, the level-crossing rate at the chosen threshold is fitted to the desired occurrence rate R_{EB} of the error bursts, where $R_{EB} = \mathcal{N}_{EB}/T_t$.

After we obtain the error burst lengths and error-free burst lengths. It is observed that the obtained lengths do not match the desired lengths properly. Therefore, two mappers are designed in order to achieve good fit to the desired burst error statistics. The number of error bursts of length m_e is denoted by $N_{EB}(m_e)$ and the number of error-free bursts of length $m_{\bar{e}}$ is denoted by $N_{EFB}(m_{\bar{e}})$. A modification to these lengths can be done such that $\tilde{N}_{EB}(m_e) = \hat{N}_{EB}(m_e)$ and $\tilde{N}_{EFB}(m_{\bar{e}}) = \hat{N}_{EFB}(m_{\bar{e}})$ hold, respectively. Here, $\hat{N}_{EB}(m_e)$ equals $\lfloor \frac{\tilde{N}_t}{N_t} N_{EB}(m_e) \rfloor$ or $\lfloor \frac{\tilde{N}_t}{N_t} N_{EB}(m_e) \rfloor +$ 1 for different error burst lengths m_e in order to fulfill $\sum_{m_e=m_{B1}}^{m_{B2}} \hat{N}_{EB}(m_e) = \tilde{\mathcal{N}}_{EB}$.

Similarly, $\hat{N}_{EFB}(m_{\bar{e}})$ equals $\lfloor \frac{\tilde{N}_t}{N_t} N_{EFB}(m_{\bar{e}}) \rfloor$ or $\lfloor \frac{\tilde{N}_t}{N_t} N_{EFB}(m_{\bar{e}}) \rfloor + 1$ for different error-free burst lengths $m_{\bar{e}}$ to satisfy $\sum_{m_{\bar{e}}=m_{\bar{B}1}}^{m_{\bar{B}2}} \hat{N}_{EFB}(m_{\bar{e}}) = \tilde{\mathcal{N}}_{EFB}$.

In order to modify the obtained error burst lengths, we first find the corresponding values $\ell^1_{m_e}$ and $\ell^2_{m_e}$ in the obtained error burst lengths to assure the following conditions

$$\sum_{l=\ell^1_{m_e}}^{\ell^2_{m_e}-1} \tilde{N}_{EB}(l) < \hat{N}_{EB}(m_e)$$

$$\text{and} \qquad \sum_{l=\ell^1_{m_e}}^{\ell^2_{m_e}} \tilde{N}_{EB}(l) \geq \hat{N}_{EB}(m_e)\ . \tag{2.7}$$

Then,

$$\sum_{l=\ell^1_{m_e}}^{\ell^2_{m_e}-1} \tilde{N}_{EB}(l) + N_{\ell^2_{m_e}} = \hat{N}_{EB}(m_e)\ . \tag{2.8}$$

The same idea applies to the obtained error-free bursts. Then the modified error bursts and error-free bursts are combined such that an error-free burst is followed by error burst and so on. Hence, an error sequence can be generated

The DPBGM is a recent and promising class of error models. It really gives a perfect match for the important burst error statistics compared with those of the reference error sequences with just a small discrepancy. More advantages for the DPBGMs are that their parameters can easily be determined, the design procedure can efficiently be implemented using the computer, and the statistical properties can be varied in a wide range. The approach has a drawback in the stage of generating error sequences, because it needs the reference error sequences to carry out the generation. In comparison, other models, especially those based on Markov chains, do not need the reference error sequences to create the error bursts and hence the error sequences.

2.3.6 Generative Models Comparison and Summary

We classified the existing generative models in the literature into five classes. Each class of models identifies a method for generating error sequences and has advantages and disadvantages. Table 2.1 summarises the differences between these different classes.

Although many improvements were accomplished to Markov models by either increasing the number of states or changing the transition conditions, all devolved models became complex with unsatisfactory results. The best model amongst Markov models is SFM as it has a good compromise between simplicity and accuracy and applied to various channels. Moreover, its error-free distributions follow a sum of exponentials. Nevertheless, SFM is not suitable to slow fading scenarios.

SFM is our choice to compare with our proposed models in addition to BWHMM and DPBGM as they are widely used. We discarded SCFG and CAGM from our comparisons as their usage is limited to specific environments and their performance can hardly conquer our selected models, namely SFM, HMM, and DPBGM.

It can also be noticed in Table 2.1 that DPBGM has the least disadvantages and the most advantages. Therefore, DPBGM is expected to take the lead amongst all generative models regarding the best performance.

Table 2.1: Generative models comparison.

Generative models	Advantages	Disadvantages
MMs	• Simply structured and easily implemented • Applied to wide range of channels	• Failed to closely match original error sequences • Have renewal natures • Parametrised based on the assumption of having geometric distribution for the run lengths
BWHMMs	• Robust and always converge	• The convergence point is not always a global maximum • BW algorithm needs long time to reach convergence points • Very complex because of the huge number of needed states • Better characterise error bursts of indoor channels
SCFGs	• More useful than BWHMM in modelling palindromes • Have faster execution than the BWHMM	• Only portray the statistics of error bursts derived from indoor environments, i.e., the error-density behaviour is bell-shaped • General performance is not better than HMM
CAGMs	• Use chaotic attractors	• Very complicated to parametrise • Take long time to execute • Do not accurately describe most of burst error statistics
DPBGMs	• Linked with the underlying fading process • Easy to understand, determine their parameters, and implement • Performance is very satisfactory • Applicable to all environments	• The generation stage is not self-driven

Chapter 3

Novel Hidden Markov Generative Models

3.1 Introduction

Markovian models have plenty of applications in many fields of science including error models. The structure of the Markovian generative models ranges from simple to very complicated. It is well-known that simplicity in modelling adversely impacts the accuracy. However, It is always very desirable to have simple models with acceptable accuracy of the outcome of the results. In this chapter, we propose a Markovian based generative models that is very simple to parametrise and the results have very satisfactory accuracy. This model is tested against hard bit and soft bit error sequences in addition to packet error sequences. Moreover, we increase the complexity of the proposed Markovian model a bit by introducing extra layer of processing but having more accurate results.

3.2 Double Embedded Processes based HMMs

In this section, a novel generative model called Double Embedded Processes based Hidden Markov Model (DEPHMM) has been proposed. This model consists of two layers of processes: the first layer is composed of only one error-free state and many error bursts states, whereas the second layer constructs detailed error bursts inside the error bursts states. Estimating the DEPHMM parameters is intuitive and plain.

3.2.1 DEPHMMs for Hard Bit-Level Error Sequences

The starting point in configuring the DEPHMM is to divide an obtained reference error sequence from a real system into error bursts and error-free bursts. But, to ascertain the error-free bursts we firstly need to find out the value of η. It can be chosen through a range of values between η_i and η_f so that the error burst identification is not affected. Both values are simply obtained from the EFRD curve when it is flat.

FIGURE 3.1 illustrates a breakdown of an error sequence. It demonstrates that an error sequences consists of error-free bursts and error bursts which in turn break into error clusters and gaps. This configuration was a source of inspiration towards constructing the DEPHMM.

FIGURE 3.2 shows a sketch of the DEPHMM. Since the error-free bursts consist of only "0"s, one state is sufficient to represent them. On the other hand, the error bursts are considered as the most important part in the DEPHMM, because they have many structural variations. Therefore, error bursts deserve to be classified into many groups. Each group should convey a specific structural behaviour. Subsequently, a structural criterion must be adopted for the purpose of error bursts classification. The maximum gap in each error burst of the reference sequence is the choice. Explicitly, the maximum gaps of error bursts which could range from 0 to $\eta - 1$ should be divided according to their histogram into the number of error burst states, where the divided intervals of the maximum gaps have approximately equal number of error bursts. By using this criterion, the model can circumscribe any degradation in error correlation, which is adequately obvious in wireless communication systems. Joining the error burst states with the error-free burst state to construct generated error sequences is the first process in the DEPHMM.

Producing error-free bursts is straightforward. Once we recognise the reference error-free burst lengths distribution, many error-free burst lengths can then be generated. The big challenge now is how to construct generated error bursts so that they convey the same behaviour of the reference error bursts. Because error bursts are composed of error clusters and gaps with lengths less than η, both of them can be represented by a separate number of states. As a result, many state configurations are possible. But, since our major apprehension is the detailed construction of errors within an error burst, we allot several substates for the error clusters of each class of the first process and single substate for the gaps of the error bursts. Each error cluster substate is occupied by a single error cluster length due to error clusters' short lengths. Similar to the error-free bursts, the production of gaps within error bursts of each class depends mainly on their length probability distribution. Connecting the error cluster substates with the gaps substate is the second process. The mean gap length of the generated error bursts in each state should match that of the original error bursts. Otherwise, the states with mismatched mean gap lengths should be further partitioned.

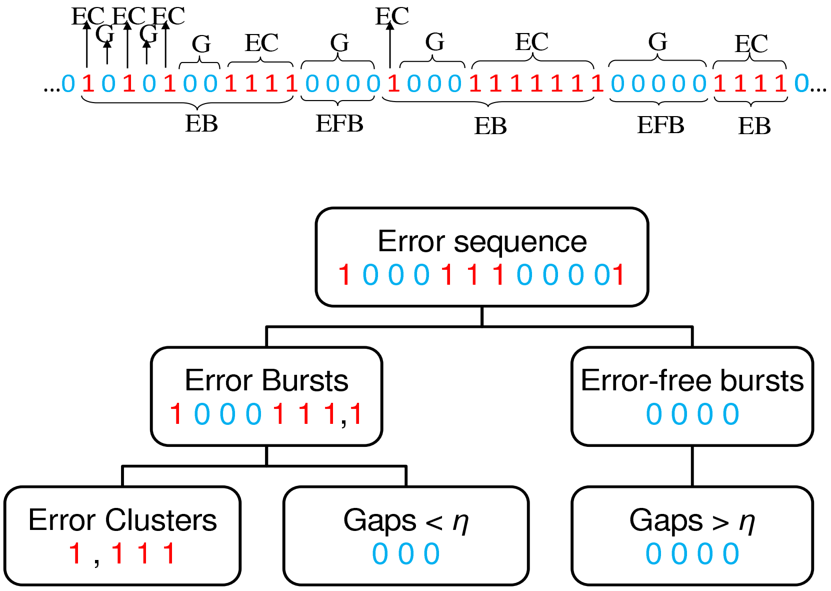

Figure 3.1: A breakdown of an extract of an error sequence.

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. N is selected according to the accuracy demand.

2) M_u: the number of error cluster substates in each state, i.e., $V_{Ou} = \{v_{1u}, v_{2u}, ..., v_{Mu}, v_{Mu+1}\}$, where V_{Ou} is the set of substates, $O = 1, ..., M$, and $u = 1, ..., N$. M_u is designated according to the number of lengths that the error clusters have in each error burst state.

3) $\mathbf{F} = (f_{i,j})$: the state transition matrix, where $f_{i,j}$ is the transition probability from State s_i to State s_j, such that

$$f_{i,j} = P\left[Q_{t+1} = s_j \,|\, Q_t = s_i\right], 1 \leq i, j \leq N + 1$$

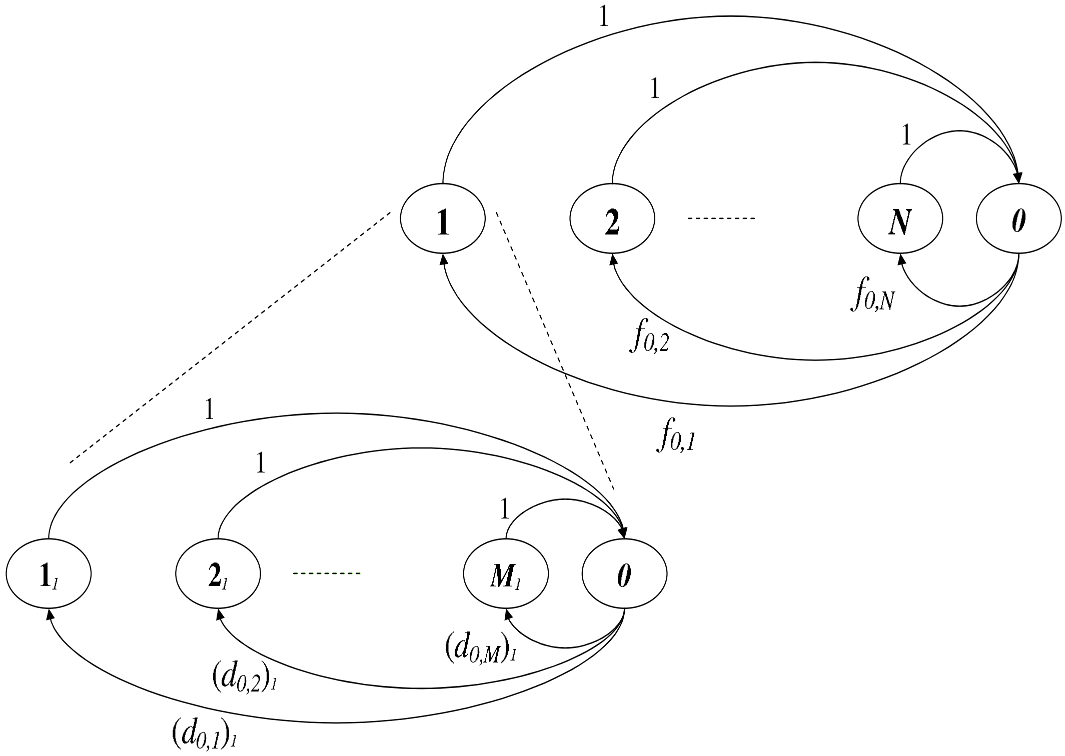

FIGURE 3.2: The DEPHMM.

$$= \begin{cases} 1, & 1 \leq i \leq N, j = N+1, \\ \frac{N_{EB,j}}{\sum_{j=1}^{N} N_{EB,j}} \approx \frac{N_{EB,j}}{N \times N_{EB,N}}, & i = N+1, 1 \leq j \leq N, \\ 0, & \text{otherwise,} \end{cases} \tag{3.1}$$

with Q_t being the current state at time t and $N_{EB,j}$ the number of error bursts in s_j. The structure of the state transition matrix is

$$\mathbf{F} = \begin{pmatrix} 0 & \cdots & 0 & 1 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 0 & 1 \\ f_{N+1,1} & \cdots & f_{N+1,N} & 0 \end{pmatrix}. \tag{3.2}$$

4) $\mathbf{D}_u = ((d_{h,k})_u)$: the substate transition matrix, where $(d_{h,k})_u$ is the transition probability from Substate v_{h_u} to Substate v_{k_u}, such that

$$(d_{h,k})_u = P\left[R_{t+1} = v_{k_u} | R_t = v_{h_u}\right], 1 \leq h_u, k_u \leq M_u + 1$$

$$= \begin{cases} 1, & 1 \leq h_u \leq M_u, k_u = M_u + 1, \\ \frac{N_{C,k_u}}{\sum_{k_u=1}^{M_u} N_{C,k_u}}, & h_u = M_u + 1, 1 \leq k_u \leq M_u, \\ 0, & \text{otherwise,} \end{cases} \tag{3.3}$$

with R_t being the current substate at time t and N_{C,k_u} the number of error clusters in Substate v_{k_u}. The structure of the substate transition matrix is

$$\mathbf{D}_u = \begin{pmatrix} 0 & \cdots & 0 & 1 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 0 & 1 \\ d_{M_u+1,1} & \cdots & d_{M_u+1,M_u} & 0 \end{pmatrix}. \tag{3.4}$$

The reason that (3.2) and (3.4) have probabilities of 0 values is that no transitions occur between the error burst states themselves as well as the error cluster substates themselves, respectively. Moreover, self transitions are not allowed. On the contrary, (3.2) and (3.4) have probabilities with values of 1. That means the transitions between the relevant states are enforced. This is natural to guarantee that each error burst is followed by a sole error-free burst, and each error cluster is followed by a single gap.

5) $\mathbf{A} = (a_j(n))$: the first process emission probability distribution matrix, where $a_j(n)$ $(1 \leq j \leq N+1)$ is the probability of getting the burst y_n in State s_j, that is

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

$N_{EFB,j}$ is the number of error-free bursts in State 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 x_m in Substate v_{k_u}, that is

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

N_{G,k_u} is the number of gaps in Substate v_{k_u}.

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

$$\mathbf{\Pi}_u = \left(d_{M_u+1,1}, ..., d_{M_u+1,M_u}, 0\right), \tag{3.5}$$

which is extremely important since it assures the initiation of an error burst by an error cluster, otherwise the definition of error burst is no longer valid. By analogy, we can obtain the initial state distribution vector.

8) $\mathbf{\Omega}_u = ((\omega_k)_u)$: the termination substate distribution vector, where $(\omega_k)_u$ is the probability of v_{k_u} to be the final substate.

$$\mathbf{\Omega}_u = \left(d_{M_u+1,1}, ..., d_{M_u+1,M_u}, 0\right), \tag{3.6}$$

which is also very important because it ensures that the error burst is finalised with an error cluster as specified in the definition.

9) $\delta_{n,u}$: error burst length values. These values regulate the termination of error bursts generation, so that Ω_u shall be activated according to them. The Activation takes place when the generated error burst lengths become 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. $\delta_{n,u}$ are acquired from the reference error burst length distribution.

10) Γ: generated error sequence length. This value terminates the error sequence generation once it is reached or exceeded. It does not matter whether the current state is error burst or error-free burst.

All the aforementioned parameters construct the DEPHMM and explain its operation.

3.2.1.1 Simulation Results and Discussions

In order to scrutinise the DEPHMM operation, a reference error sequence needs to be obtained for its parametrisation. To achieve this, we consider an uncoded EGPRS transmission system with ideal frequency hopping. The elementary digital channel is constituted of a Gaussian minimum shift keying (GMSK) modulator, an interfered

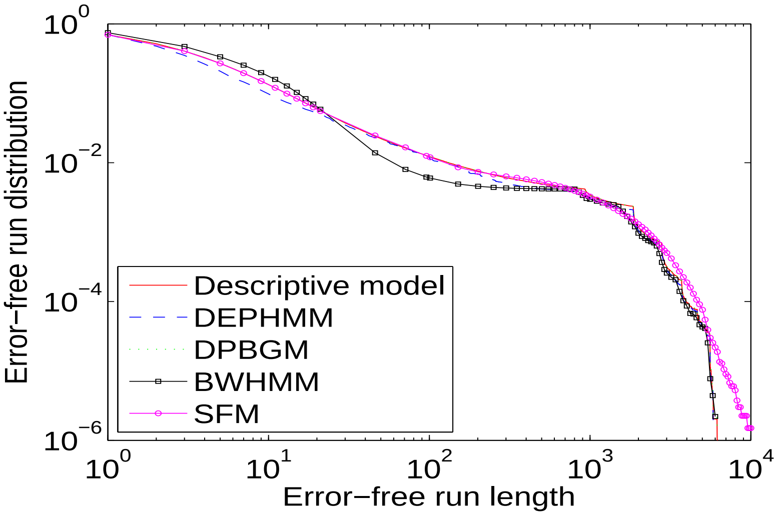

Figure 3.3: EFRDs of the descriptive model obtained from the EGPRS system and different generative models.

propagation channel, a GMSK demodulator, and a hard decision Viterbi equaliser [77]. The underlying channel is tailored to TU environment with CIR of 8 dB, and mobile speed of 3 km/h. The data are transmitted using time-division multiple access (TDMA) with blocks of 116 bits and a transmission rate of $F_s = 270.8$ kb/s

The reference error sequence has 15 million bits. It exhibits long error bursts interweaved with long error-free bursts. It has 4269 error bursts and 4268 error-free bursts with maximum lengths of 6489 and 6251 bits, respectively. We find η from FIGURE 3.3 which displays the EFRD. From its shape plateau, we find $\eta_i = 400$ and $\eta_f = 1000$ hold. Here, η_i is the beginning of the plateau and η_f is the end of the plateau. The selection of any value η between η_i and η_f results in the same burst identification because the probability of encountering error-free bursts of lengths ranging from η_i to η_f is negligible [45]. The chosen value of η is 800. $\Gamma = 21$ million bits.

For the sake of comparison, a SFM, BWHMM, and DPBGM have been implemented. The parameters of a SFM with N states are obtained by fitting the weighted sum of $N - 1$ exponentials to the EFRD. The number of states used for the SFM is 6. In fact, no better improvement of the SFM statistics could be attained by increasing its number of states to more than 6. Therefore, this number of states makes SFM to

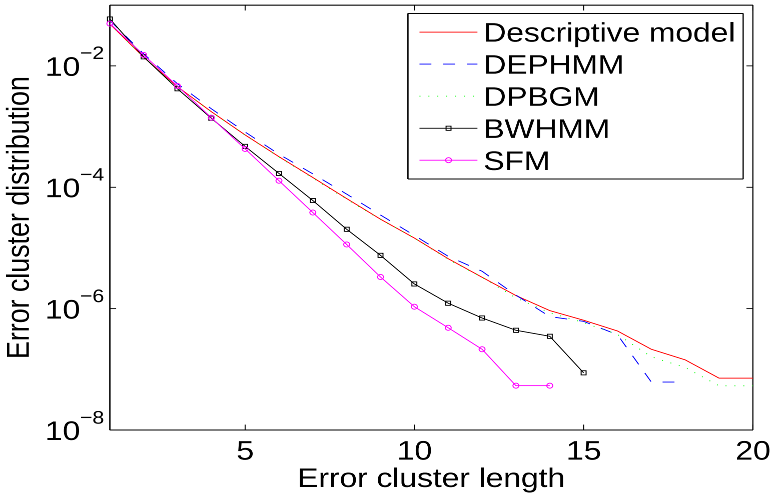

FIGURE 3.4: ECDs of the descriptive model obtained from the EGPRS system and different generative models.

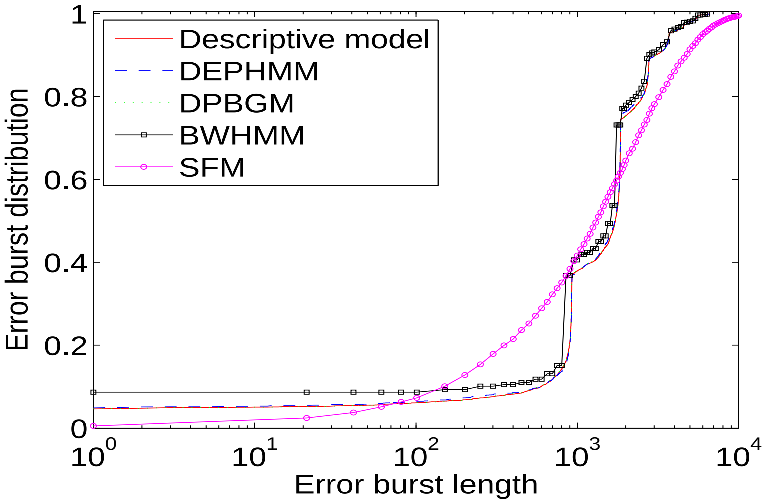

FIGURE 3.5: EBDs of the descriptive model obtained from the EGPRS system and different generative models.

compromise between accuracy and complexity, such that adding more states increases the complexity whilst no further enhancement to the SFM performance occur. On the contrary, reducing the number of states to less than 6 reduces the accuracy. In the BWHMM, the number of classes is 12, whereas the total number of states is 400. The number of bits which should represent each block in the error bursts is

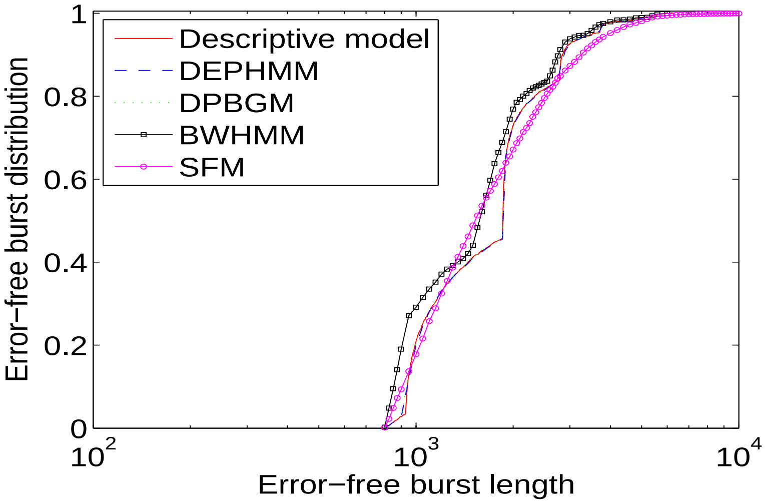

FIGURE 3.6: EFBDs of the descriptive model obtained from the EGPRS system and different generative models.

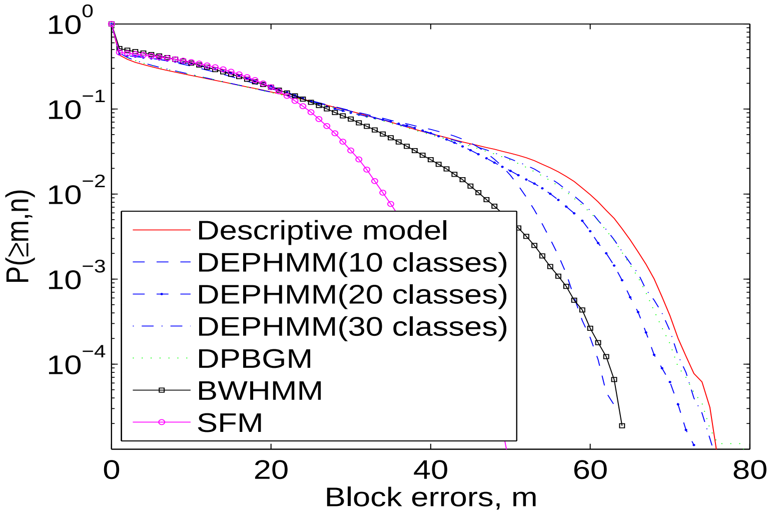

FIGURE 3.7: BEPDs of the descriptive model obtained from the EGPRS system and different generative models ($n = 116$).

chosen to be 103 bits. For the DPBGM, the vector $\mathbf{\Psi}$=($N_1 = 9, N_2 = 10, r_{th} = 0.09, \sigma_0 = 0.0783, f_{max} = 73.22Hz, T_A = 0.8132ms$), such that N_1 and $N_2 \geq 7$ in order to compare the deterministic process statistics with Rayleigh process statistics [70], $r_{th} \ll 1$ (any value below 1 works because of the usage of the mappers). Here also σ_0, f_{max}, and T_A are obtained from (2.4), (2.5), and (2.6), respectively, whilst $\mathcal{R}_B =$

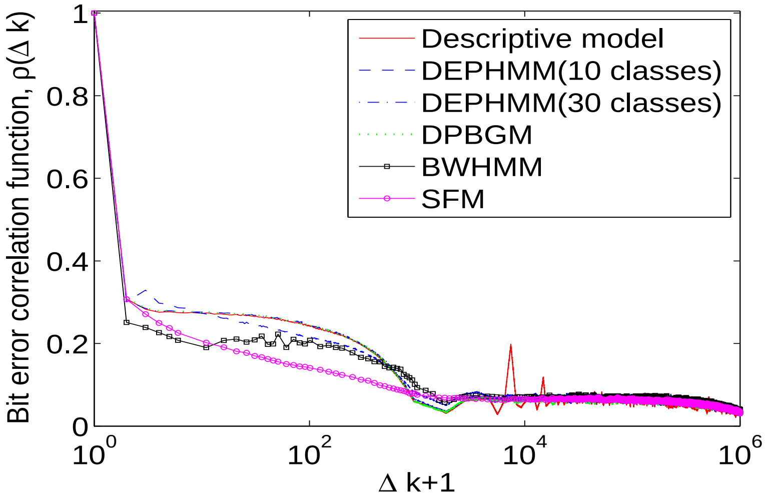

FIGURE 3.8: BECFs of the descriptive model obtained from the EGPRS system and different generative models.

0.9344, and $q_s = 0.01$.

To assess any generative model, we need to investigate how close its burst error statistics can match the descriptive model. FIGURES 3.3–3.8 illustrate the behaviour of the burst error statistics, mentioned in Subsection 2.2.1, for the descriptive model and the well-known generative models addressed before. It is apparent that the SFM fails to characterise all the statistics except the EFRD, since the SFM is designed by fitting it. The BWHMM depicts an enhancement to the SFM, despite the fact that it misses the contiguity with the descriptive model. This is because the BWHMM was designed to best describe error sequences with bell-shaped error density bursts. But, our reference error sequence has many error bursts which do not comply with such a shape. However, the DPBGM statistics has small differences from the DEPHMM statistics of high accuracy. Nevertheless, the DEPHMM leads the other generative models since the DPBGM fetches the error bursts from the reference error sequence rather than constructing them by itself.

In FIGURES 3.3–3.6, 10 classes of error bursts are selected for the DEPHMM. However, this number of classes is not sufficient to fit the EFBD and BECF in FIGURES 3.7–3.8 perfectly. Therefore, we increased this number to 20 and 30. As a result, a notable augmentation in the performance is achieved. Although the total number of states of the DEPHMM reaches 260 (including the substates), it is still less than the number of states of the BWHMM. Similar to the EFBD and BECF, it is worth mentioning that the burst error statistics in FIGURES 3.3–3.6 could be enhanced as well by increasing the number of classes.

3.2.2 DEPHMMs for Soft Bit-Level Error Sequences

Dealing with soft error sequences is quite similar to hard error sequences when parameterising the DEPHMM. We start with decomposing an obtained soft reference error sequence into soft error bursts and soft error-free bursts. However, to extract the error-free bursts the value of η must be determined. It can be chosen from a range of values between η_i and η_f so that the soft error burst identification is not seriously affected. Both values are simply obtained from the SEFRD curve when it becomes approximately flat.

Again, one state is used to represent the soft error-free bursts in the first process because it contains nonnegative integers only. Whereas, the error bursts are more complex in the DEPHMM than the soft error-free bursts, because they have negative and nonnegative integers arranged variously. Therefore, soft error bursts deserve partitioning into many groups. Each group should convey a specific structural behaviour. Subsequently, a structural criterion must be adopted for partitioning the soft error bursts. Since the errors (nonnegative integers) control the behaviour of the soft error sequence, we have decided to use the number of soft error clusters as a criterion. However, the length of the soft error bursts varies and this would affect the number of soft error clusters. Therefore, the ratio of the number of soft error clusters in an error burst to the length of the error burst is the best choice for the purpose of partitioning. Notice that this criterion of division is different from the one of the hard bit DEPHMM. The range of ratios should then be divided into the number of soft error burst states, where each state has approximately the same numbers of soft error bursts. By utilising this criterion, the model can delimit serious degradation in the error correlation, which is obvious in wireless communication systems. Combining the soft error burst states with the soft error-free burst state to produce error sequences is the task of the first process in the DEPHMM.

Generating soft error-free bursts' lengths is straightforward. Once we find the reference error-free burst lengths distribution, many error-free burst lengths can then be generated by knowing the distribution of each integer from the reference error sequence. The remaining challenge is how to create soft error burst lengths so that they convey similar statistics to the reference error bursts. Since soft error bursts are composed of soft error clusters as well as soft gaps with lengths less than η, both of them are represented by a number of states. Therefore, several substates have been selected for the soft error cluster lengths of each class of the first process and a single substate for the soft gap lengths of the error bursts. Each error cluster substate is alloted a single error cluster length because of error clusters' short lengths. Similar to the soft error-free bursts, the generation of soft gaps lengths within the soft error bursts of each class depends basically on their length probability distribution. Connecting the soft error cluster substates with the soft gaps substate is certainly the second process. Thus, Figure 3.2 applies.

The percentage of error numbers for each state of error burst states should approximately match the one of the reference error sequence. However, if the percentage is much larger than the one of the reference sequence, then the state should be further divided. On the contrary, if the percentage is much less than the one of the reference sequence, then 10-50% of the soft gap lengths must be deleted till the previous condition is accomplished.

At the same time, the distribution of each symbol must be known for each state in the reference error sequence. Subsequently, the symbols of the soft gaps and error clusters are substituted in the generated error bursts and error-free bursts according to their reference distribution. A prior knowledge of the number of generated error bursts to achieve a target length of generated error sequence is helpful to keep similar generated symbol distribution to the reference error sequence. The parameters of the soft bit DEPHMM are very similar to those of the hard bit DEPHMM.

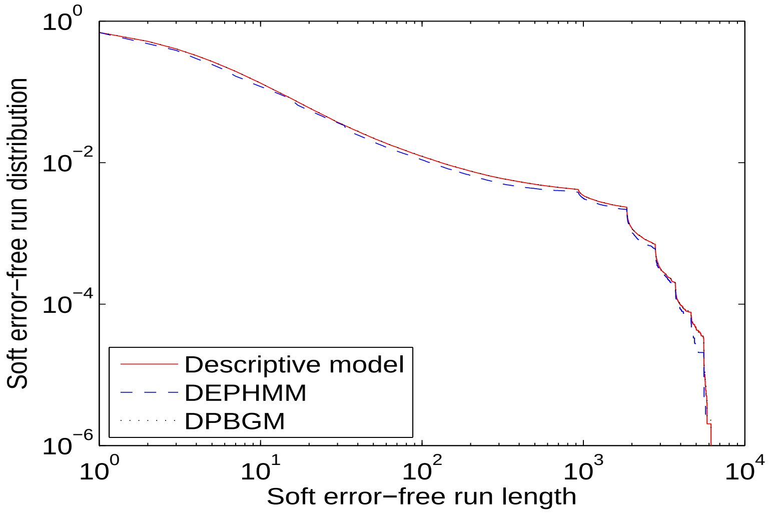

Figure 3.9: SEFRDs of the descriptive model obtained from the EGPRS system and different generative models.

3.2.2.1 Simulation Results and Discussions

In order to examine the ability of the soft DEPHMM to produce satisfactory results of burst error statistics, a reference soft error sequence is obtained for the purpose of parametrisation. Therefore, we employ the same EGPRS transmission system with small difference. Here, the underlying discrete channel is constructed from a GMSK modulator, an interfered propagation channel, a GMSK demodulator, and a 4-bit soft decision Viterbi equaliser. The underlying channel is still tailored to a TU environment with CIR of 8 dB, and mobility speed of 3 km/h.

The reference soft error sequence has 10 million integers, whereas $\Gamma = 20$ million bits. We find η from FIGURE 3.9 which displays the SEFRD. From the shape of its plateau, the best value of η is 800. Afterwords, the soft error bursts can be extracted and partitioned into classes. The number of partitions should be high e.g., more than 10. The choice of this number is a trade-off between the accuracy of the final results and the complexity of the model [51]. For our soft DEPHMM, we choose the number of states to be 20. For the purpose of comparison, a soft DPBGM has been implemented. The parameters of the soft DPBGM are the same as those of hard bit error sequences.

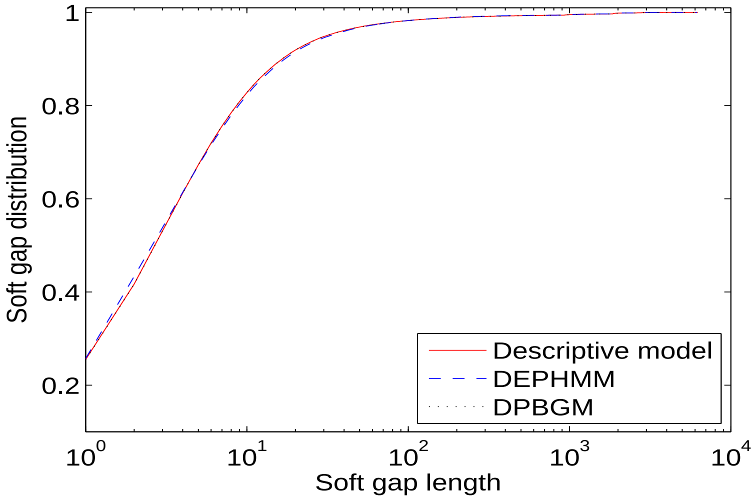

FIGURE 3.10: SGDs of the descriptive model obtained from the EGPRS system and different generative models.

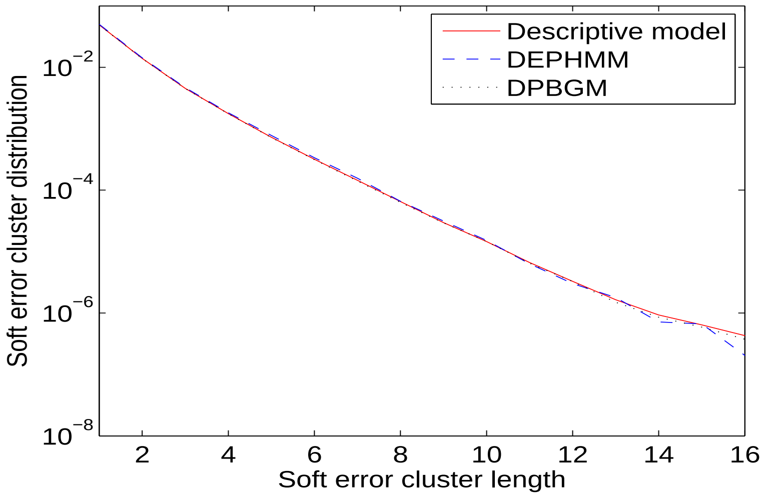

FIGURE 3.11: SECDs of the descriptive model obtained from the EGPRS system and different generative models.

FIGURES 3.9–3.14 illustrate the behaviour of the soft burst error statistics for the descriptive model and the DPBGM. The DPBGM burst error statistics have small differences from the DEPHMM statistics. Nevertheless, the DEPHMM leads the DPBGM since the latter retrieves the soft error bursts from the reference soft error sequence rather than constructing them by itself. In FIGURE 3.9, the SEFRD is

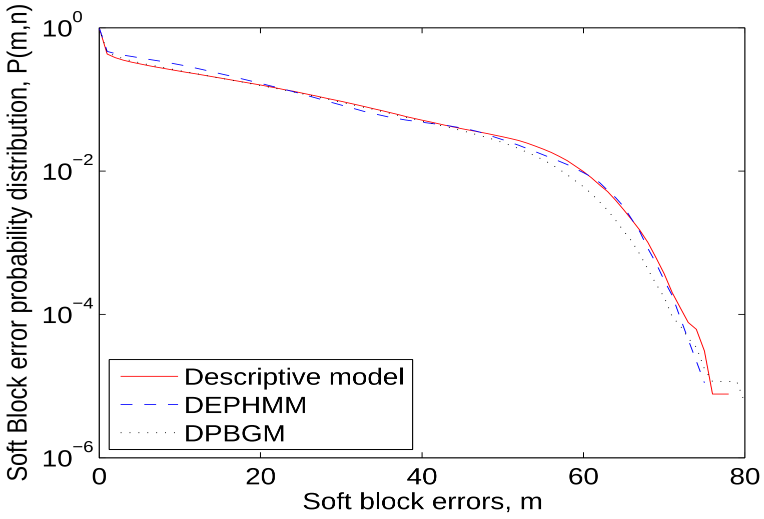

FIGURE 3.12: SBEPDs of the descriptive model obtained from the EGPRS system and different generative models.

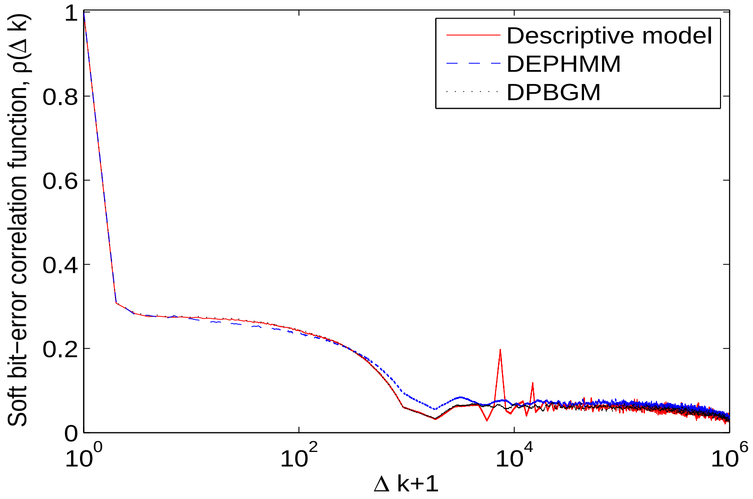

FIGURE 3.13: SBECFs distributions of the descriptive model obtained from the EGPRS system and different generative models.

drawn, where the DEPHMM has very small difference from the descriptive model. This difference is negligible. The SGD in FIGURE 3.10 almost matches the one of the descriptive model. FIGURE 3.11 shows the SECD which has a little deficiency at large lengths of soft error clusters. This is because the probabilities to select these lengths are smaller than the others in the second process of the DEPHMM. The SBEPD is

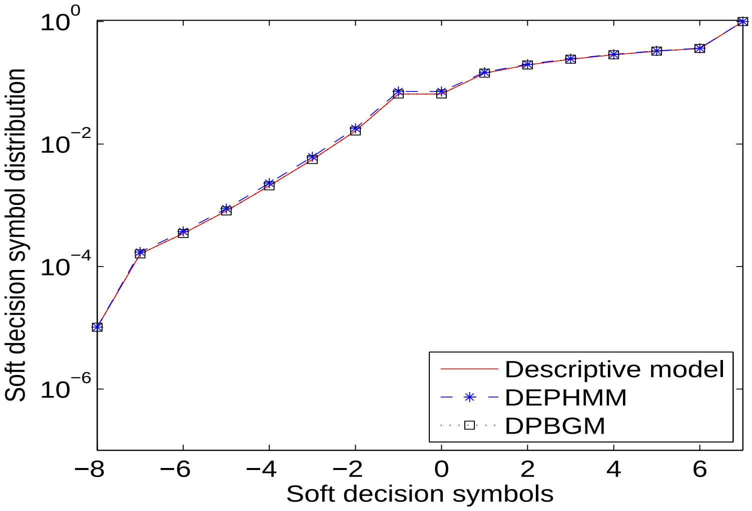

FIGURE 3.14: SDSDs of the descriptive model obtained from the EGPRS system and different generative models.

depicted in FIGURE 3.12. This statistic is very important to evaluate and design many error coding schemes. The result of this statistic for the DEPHMM is satisfactory compared to the DPBGM, the value of $n = 116$ holds. In FIGURE 3.13, the SBECF is illustrated. It has a sufficient image of the descriptive model as well. The SBECF is vital to the evaluation and designation of bit interleavers. Finally, FIGURE 3.14 shows the SDSD. Here, the soft decision has $M = 4$, therefore, $e_j \in [-8, +7]$. This statistic shows a competent match between the DEPHMM and the descriptive model.

3.2.3 DEPHMMS for Packet-Level Error Sequences

The steps for formulating the packet DEPHMM is similar to those of the hard bit DEPHMM because both of them require binary error sequences, Hence, FIGURE 3.2 applies. However, partitioning the packet error bursts has here different criterion from the hard bit DEPHMM. The ratio of the number of error packets in an error burst to the length of the error burst has been used as a criterion for 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. By utilising this criterion, the model can delimit serious degradation in the error packet correlation, which is perceived in packet error sequences.

The parameters of the packet DEPHMM are almost the same of those of the hard bit DEPHMM.

3.2.3.1 Simulation Results and Discussions

In order to obtain a reference packet error sequence for the purpose of parametrisation, we simulate again an uncoded EGPRS transmission system with ideal frequency hopping. However, the CIR value is changed to 11 dB. The length of the reference packet error sequence is 1 million. It 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.

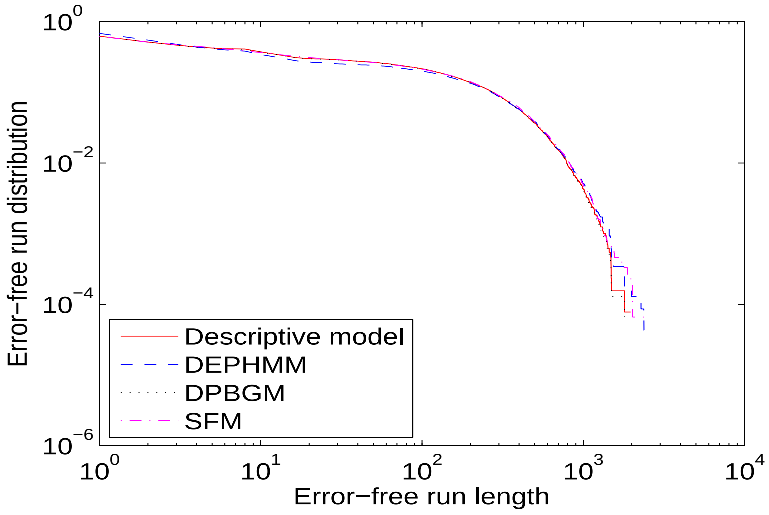

FIGURE 3.15: EFRDs of the descriptive model obtained from the EGPRS system and different generative models.

The value of η is chosen to be 50 from the EFRD plateau of FIGURE 3.15 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. The selected number of states $N = 20$. For the purpose of comparison, a packet DPBGM and SFM are

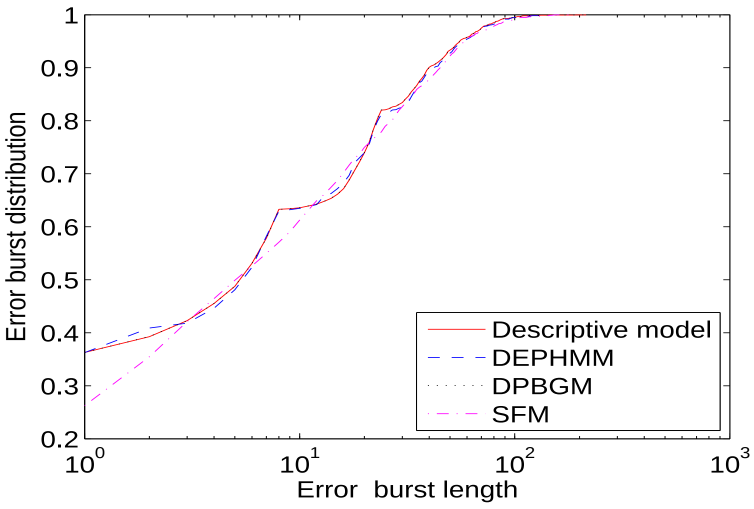

FIGURE 3.16: EBDs of the descriptive model obtained from the EGPRS system and different generative models.

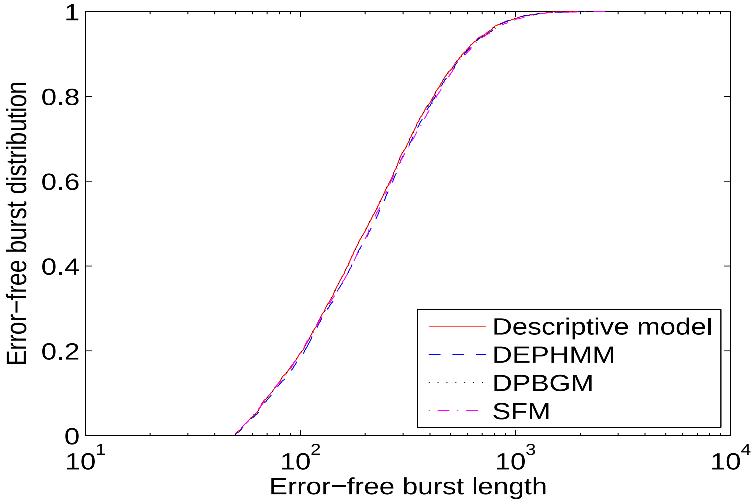

FIGURE 3.17: EFBDs of the descriptive model obtained from the EGPRS system and different generative models.

implemented. The parameters of the packet DPBGM are $\mathcal{R}_B = 0.052$ and $q_s = 0.01$. Subsequently, the vector $\mathbf{\Psi} = (9, 10, 0.09, 0.2255, 300.85Hz, 0.0394ms)$, and the number of states used for the SFM is 6.

For the purpose of evaluating the packet DEPHMM, we need to find out how closely

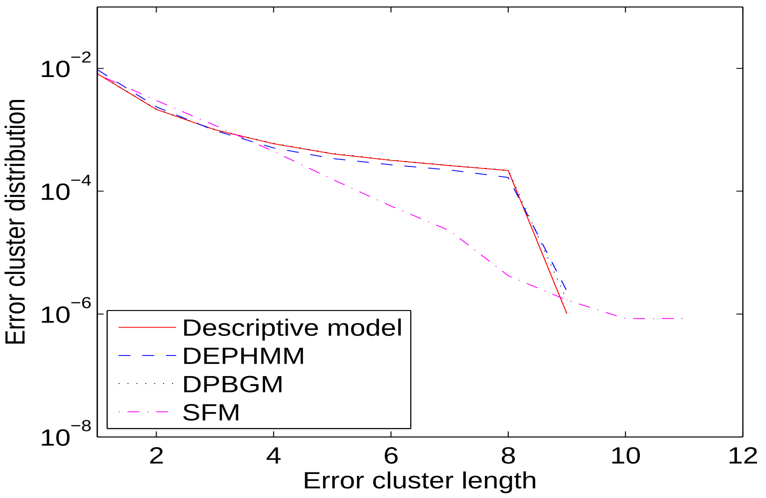

FIGURE 3.18: ECDs of the descriptive model obtained from the EGPRS system and different generative models.

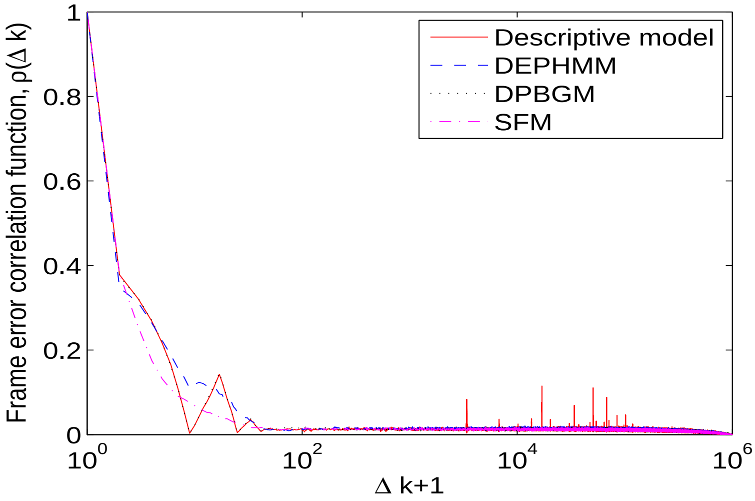

FIGURE 3.19: PECFs of the descriptive model obtained from the EGPRS system and different generative models.

its burst error statistics can match the descriptive model. FIGURES 3.15–3.19 demonstrate the performance of the packet DEPHMM, DPBGM, and SFM compared with the descriptive model in terms of the burst error statistics. It is clear that the DEPHMM burst error statistics have small discrepancies from the DPBGM statistics and the descriptive model. The high performance quality for the DPBGM is due to the

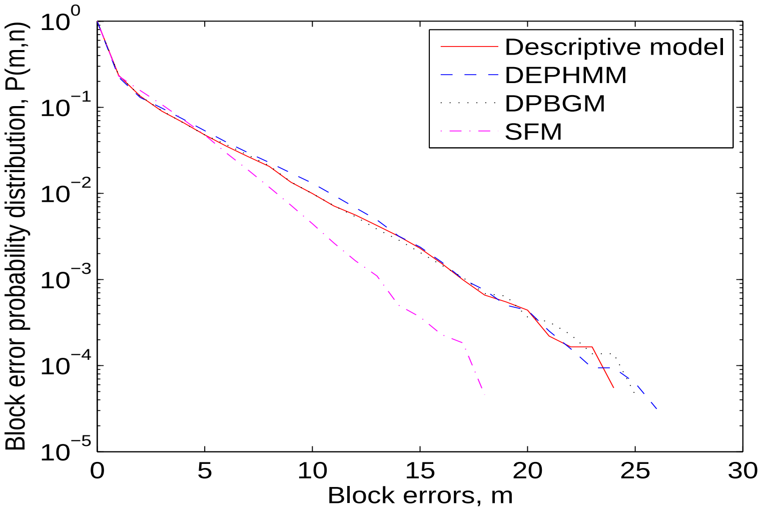

FIGURE 3.20: BEPDs of the descriptive model obtained from the EGPRS system and different generative models.

borrowed error bursts from the reference packet error sequence. Nevertheless, this borrowing process is not desirable in generative models. Therefore, the DEPHMM can be considered better than the DPBGM. Figure 3.15 depicts the EFRD where the DEPHMM has very small difference from the descriptive model. This difference is negligible. The EBD of the DEPHMM in Figure 3.16 almost matches that of the descriptive model. Figure 3.17 shows the EFBD of the DEPHMM with good agreement with the descriptive model. The ECD of the DEPHMM is illustrated in Figure 3.18, it has a little divergence for long error clusters, e.g., 8. This is because the probability of selecting these lengths are smaller than the others in the second process of the DEPHMM. In Figure 3.20, 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 so it shows average correlation values. Finally, The BEPD is illustrated in Figure 3.19, the value of $n = 55$ is selected. This statistic shows a good match between the DEPHMM and the descriptive model. Whereas, the SFM failed to match most of the burst error statistics except the EFRD.

3.3 Layered HMMs

Hierarchical [78–80] and Layered HMMs (LHMMs) [81] have been introduced to dramatically reduce processing the amount of training data required for them to achieve a comparable performance to conventional HMMs.

The LHMM is a statistical model derived from the HMM. A LHMM consists of several levels of HMMs, where the HMMs on level $i + 1$ correspond to observation symbols or probability generators at level i. Every level i of the LHMM consists of K_i HMMs running in parallel [82].

In this section, we propose a novel generative model, called three layered HMMs (3LHHM). The first layer is made up of one error-free burst state and several error burst states assigned according to the maximum error cluster lengths. The second layer divides further the classes of error bursts based on their maximum gap lengths.

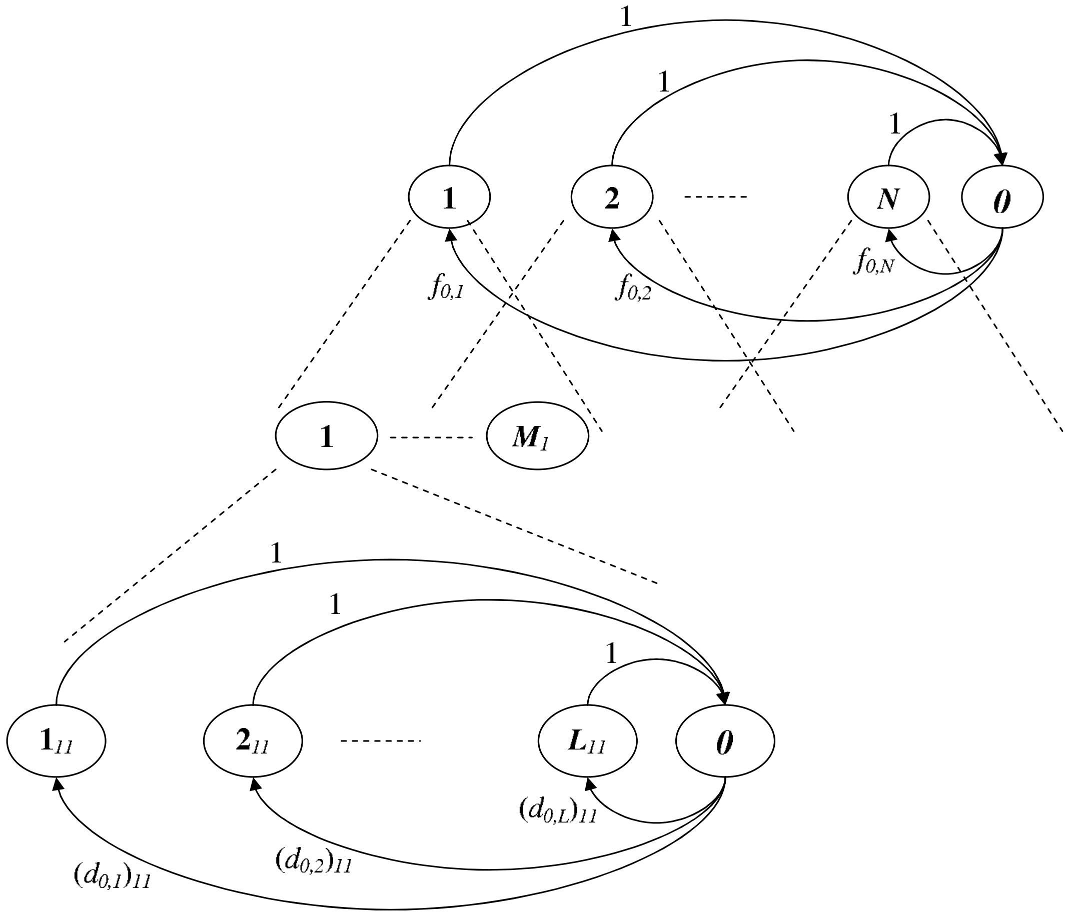

FIGURE 3.21: The 3LHMM.

The final layer constructs the error bursts. Implementing the 3LHMMs and determining its parameters are simple and straightforward. Binary bit error sequences are used and the resulting burst error statistics are competent.

3.3.1 Three Layered HMMs for Hard Bit Error Sequences

In order to distinguish the error-free bursts from the error bursts, the value of η should be determined. As we have previously learnt, the value of η can be obtained from the EFRD curve when it is flat.

The 3LHMM design is shown in FIGURE 3.21. One state is used to represent the error-free bursts because they consist of "0"s only. On the contrary, the error-bursts have various structural configurations and therefore are entitled to classification. Thus, several error burst states are required, each state typifies a common structural behaviour. The classification criterion for the first layer is the maximum error cluster length in each error burst. This means that each class is represented by one state and each state contains all error bursts that have the same maximum error cluster length. The number of values of the maximum error cluster lengths in the original error sequence determines the number of error burst states in the first layer. For the purpose of further classification, each error burst state in the first layer can be partitioned into internal states according to the maximum gap lengths of its error bursts. The internal states have error bursts within equal intervals of the entire maximum gap lengths range of each error burst state. The maximum gap lengths in each state have long range, consequently, the number of internal states can be large, e.g., 8-15. Those internal states are held to be the second layer in the model. The third layer is dedicated to constructing error bursts. Clearly, an error burst consists of error clusters of different lengths and gaps of lengths less than η. Since the errors are our concern in error models, we allocate several substates of the second layer for the error clusters in a manner that each substate represents one error cluster length. Moreover, because the gaps of the error bursts consist of "0"s only, one state is still sufficient to represent them. Generating gaps of length less than η in the third layer and error-free bursts in the first layer depends only on their lengths distribution.

The parameters of the 3LHMM are as follows:

1) N: the number of error burst states in the first layer, i.e., $S = \{s_1, s_2, ..., s_N, s_{N+1}\}$, where S is the set of states and N is the number of the values of the maximum error cluster lengths in the original error sequence.

2) M_p: the number of the internal states for each error burst state, i.e., $W_{O_p} = \{w_{1_p}, w_{2_p}, ..., w_{M_p}\}$, where W_{O_p} is the set of internal states, $O = 1, ..., M$, and $p = 1, ..., N$. The parameter M_p is chosen to be large enough e.g., 8-15.

3) $L_{p,q}$: the number of error clusters substates in each internal state, i.e., $V_{p,q} = \{v_{1_{p,q}}, v_{2_{p,q}}, ..., v_{L_{p,q}}, v_{L_{p,q}+1}\}$, where $V_{p,q}$ is the set of substates and $q = 1, ..., M_p$.

The parameter $L_{p,q}$ is designated according to the number of error cluster lengths in each internal state, such that each length has one substate.

4) $\mathbf{F} = (f_{i,j})$: the state transition matrix, where $f_{i,j}$ is the transition probability from State s_i to State s_j, such that

$$\begin{aligned} f_{i,j} &= P\left[Q_{t+1} = s_j \,|\, Q_t = s_i\right], 1 \leq i, j \leq N+1 \\ &= \begin{cases} 1, & 1 \leq i \leq N, j = N+1, \\ \frac{N_{EB,j}}{\sum_{j=1}^{N} N_{EB,j}}, & i = N+1, 1 \leq j \leq N, \\ 0, & \text{otherwise}, \end{cases} \end{aligned} \tag{3.7}$$

with Q_t being the current state at time t and $N_{EB,j}$ the number of error bursts in State s_j. The state transition matrix is

$$\mathbf{F} = \begin{pmatrix} 0 & \cdots & 0 & 1 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 0 & 1 \\ f_{N+1,1} & \cdots & f_{N+1,N} & 0 \end{pmatrix}. \tag{3.8}$$

5) $\mathbf{D}_{p,q} = ((d_{h,k})_{p,q})$: the substate transition matrix, where $(d_{h,k})_{p,q}$ is the transition probability from Substate $v_{h_{p,q}}$ to Substate $v_{k_{p,q}}$, such that

$$\begin{aligned} (d_{h,k})_{p,q} &= P\left[R_{t+1} = v_{k_{p,q}} | R_t = v_{h_{p,q}}\right], 1 \leq h_{p,q}, k_{p,q} \leq L_{p,q} + 1 \\ &= \begin{cases} 1, & 1 \leq h_{p,q} \leq L_{p,q}, k_{p,q} = L_{p,q} + 1, \\ \frac{N_{C,k_{p,q}}}{\sum_{k_{p,q}=1}^{L_{p,q}} N_{C,k_{p,q}}}, & h_{p,q} = L_{p,q} + 1, 1 \leq k_{p,q} \leq L_{p,q}, \\ 0, & \text{otherwise}, \end{cases} \end{aligned} \tag{3.9}$$

with R_t being the current state at time t and $N_{C,k_{p,q}}$ the number of error clusters in Substate $v_{k_{p,q}}$. The substate transition matrix is

$$\mathbf{D}_{p,q} = \begin{pmatrix} 0 & \cdots & 0 & 1 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & 0 & 1 \\ d_{L_{p,q}+1,1} & \cdots & d_{L_{p,q}+1,L_{p,q}} & 0 \end{pmatrix}. \tag{3.10}$$

6) $\mathcal{A} = (a_j(n))$: the first layer emission probability distribution matrix, where $a_j(n)$ $(1 \leq j \leq N+1)$ is the probability of getting the burst x_n in State s_j, that is

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

$N_{EFB,j}$ is the number of error-free bursts in State s_j.

7) $\mathcal{B} = (b_u(m))$: the second layer emission probability distribution matrix, where $b_u(m)$ is the probability of getting the error burst y_m from the internal State w_{1_p}, that is

$$b_u(m) = P\left[y_m \,\mathrm{at}\, t | T_t = w_{1_p}\right], 1 \leq m \leq M_p,$$

where T_t is the current state at time t.

8) $\mathcal{E} = (e_{k_{p,q}}(\ell))$: the third layer gap emission probability distribution matrix, where $e_{k_{p,q}}(\ell)$ $(k_{p,q} = L_{p,q}+1)$ is the probability of getting the gap z_ℓ in Substate $v_{k_{p,q}}$, that is

$$e_{k_{p,q}}(\ell) = P\left[z_\ell \,\mathrm{at}\, t | R_t = v_{k_{p,q}}\right], 1 \leq \ell \leq N_{G,k_{p,q}}.$$

$N_{G,k_{p,q}}$ is the number of gaps in Substate $v_{k_{p,q}}$.

9) $\mathbf{\Pi}_{p,q} = ((\pi_k)_{p,q})$: the initial substate distribution vector, where $(\pi_k)_{p,q}$ is the probability of Substate $v_{k_{p,q}}$ to be an initial substate.

$$\mathbf{\Pi}_{p,q} = \left(d_{L_{p,q}+1,1}, ..., d_{L_{p,q}+1,L_{p,q}}, 0\right), \tag{3.11}$$

which assures the initiation of an error burst by an error cluster, otherwise the definition of error burst is no longer valid. Similarly, we can obtain the termination substate distribution vector $\mathbf{\Omega}_{p,q}$, which ensures that the error burst is ending with an error cluster, and the initial state distribution vector.

10) $\delta_{n,p,q}$: error burst length values. These values regulate the termination of error burst lengths, so that $\mathbf{\Omega}_{p,q}$ shall be activated according to them. The activation takes place when the generated error burst lengths become either equal or around a chosen δ_n. The deviation from δ_n shall be small enough, otherwise, the current generated error burst shall be discarded and the process shall be repeated again. The $\delta_{n,p,q}$ are acquired from the reference error burst lengths distribution.

11) Γ: the length of the generated error sequence. This value terminates the error sequence generation once it is reached or exceeded regardless of the current state, i.e., whether it is error burst or error-free burst state.

The above mentioned parameters set up the 3LHMM. In summary, each substate in the third layer of the 3LHMM creates error bursts and forward them to the second layer, which in turn collects all the created error bursts and forward them to the first layer. The first layer combines the error bursts with the error-free bursts to produce error sequences. But, before generating error sequences, the produced error bursts in each state should be tested to ensure that they convey similar statistical behaviour to the originals. The testing parameters in each state are the mean value of error cluster lengths and mean value of the gap lengths of the error bursts. If the matching test fails, it can be conducted to the internal states of the failed state. The internal state which fails the test must be divided into several parts. If this procedure does not work, then 10-50 % of the related gaps in the failed internal state should be deleted since many of them are duplicate.

3.3.1.1 Simulation Results and Discussions

For the sake of model parametrisation, a reference error sequence should be obtained from a real system as usual. We have used the same EGPRS transmission system which is aforementioned in Subsection 3.2.1.1 to get bit error sequences.

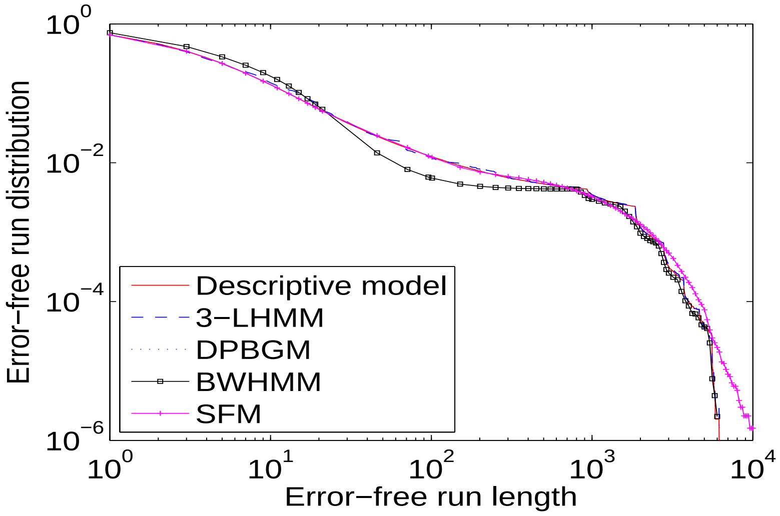

FIGURE 3.22: EFRDs of the descriptive model obtained from the EGPRS system and different generative models.

The chosen reference error sequence has 15 million bits and is corresponding to a CIR of 8 dB. It exhibits long error bursts interleaved by long error-free bursts. We find η from FIGURE 3.22 to be 800. The value of N is 19, whereas M_p is fixed to 10 and $\Gamma = 20$ million bits.

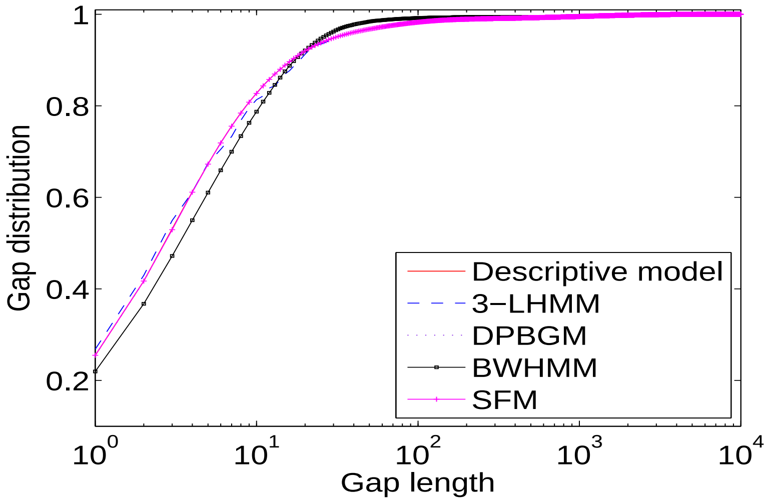

FIGURE 3.23: GDs of the descriptive model obtained from the EGPRS system and different generative models.

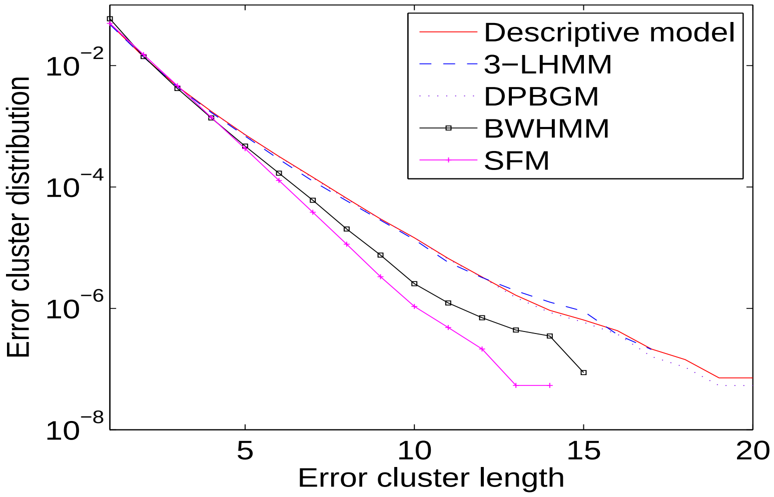

FIGURE 3.24: ECDs of the descriptive model obtained from the EGPRS system and different generative models.

For the purpose of comparison, again a SFM, BWHMM, and DPBGM are implemented. The number of states used for the SFM is 6. In the BWHMM, the number of classes is 12, whereas the total number of states is 400. The number of bits which should represent each block in the error sequence is chosen to be 103 bits.

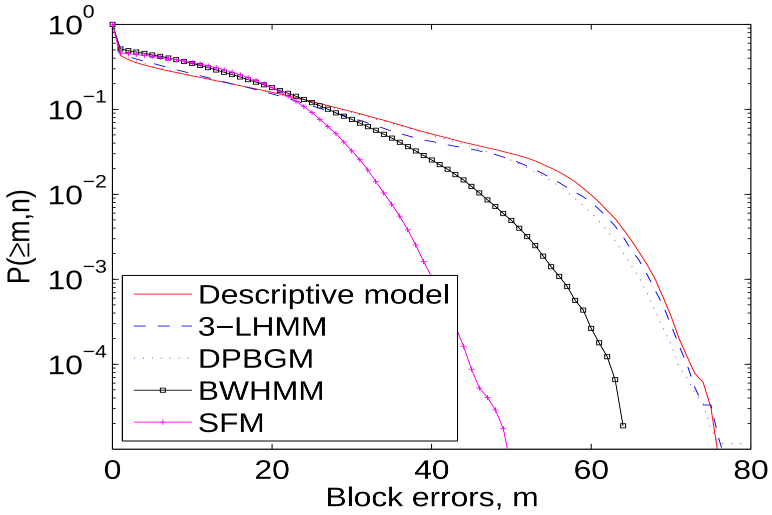

FIGURE 3.25: BEPDs of the descriptive model obtained from the EGPRS system and different generative models.

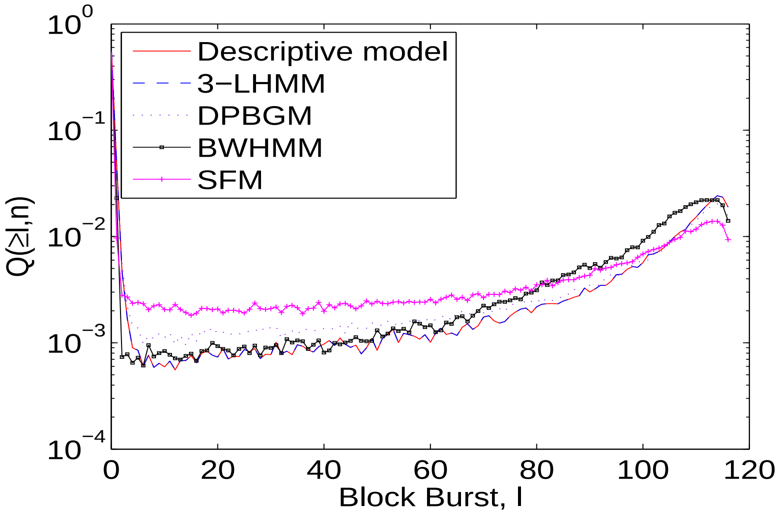

FIGURE 3.26: BBPDs distributions of the descriptive model obtained from the EGPRS system and different generative models.

For the DPBGM, the vector $\mathbf{\Psi}$ =($N_1 = 9, N_2 = 10, r_{th} = 0.09, \sigma_0 = 0.0783, f_{max} = 73.22Hz, T_A = 0.8132ms$), such that N_1 and $N_2 \geq 7$ in order to compare the deterministic process statistics with Rayleigh process statistics [70], $r_{th} \ll 1$ (any value below 1 works because of the usage of the mappers). Here also σ_0, f_{max}, and T_A are obtained from (2.4), (2.5), and (2.6), respectively, whilst $\mathcal{R}_B = 0.9344$, and $q_s = 0.01$.

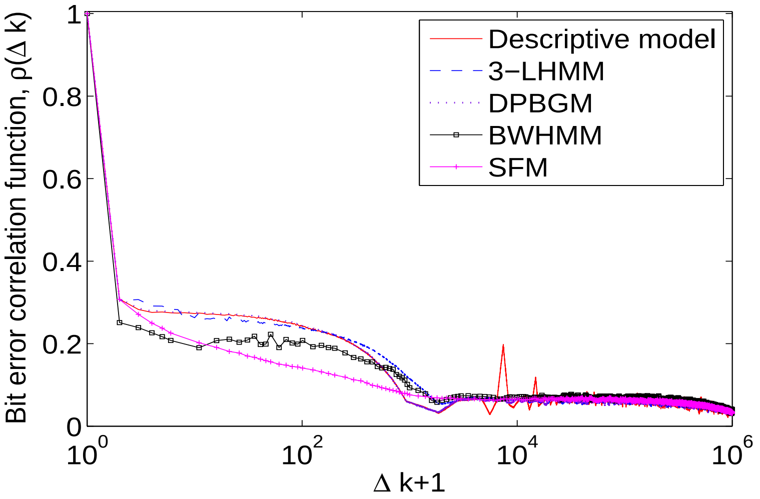

FIGURE 3.27: BECFs of the descriptive model obtained from the EGPRS system and different generative models.

In order to appraise the 3LHMM, we have to inspect how close its burst error statistics can fit those of the reference error sequence. FIGURES 3.22–3.27 demonstrate the behaviour of the burst error statistics of an EGPRS reference error sequence and those attained from the generative models. It is shown that the SFM fails to describe the ECD, BEBD, BBPD, and BECF statistics, whereas the BWHMM has better description for them than the SFM except for the EFRD and GD. On the other hand, the BWHMM burst error statistics performance still does not reach the one of the reference error sequence. This remark excludes the BBPD which has good behaviour for block bursts of lengths 0-60 bits. The lack of agreement is because that the BWHMM behaviour was conceived to best characterise the bell-shaped error density bursts. But, our reference error sequence has many error bursts which do not conform to such a shape. However, the EFRD, GD, ECD, BEBD, BBPD, and BECF statistics of the DPBGM have small differences from the 3LHMM statistics which nearly match the reference sequence statistics. On the other hand, the 3LHMM illustrates perfect match to the reference sequence for the BBPD. Therefore, the 3LHMM leads the other generative models given that the DPBGM retrieves the error bursts from the reference error sequence instead of constructing them by itself.

3.4 Summary

Having accurate error models with not very complicated processing and parametrisation is strongly desirable. This can be achieved by implementing the double embedded processes based HMM and the layered HMM. Simulation results have shown that the output burst error statistics match the desired statistics perfectly. Therefore, the performance of the proposed models is satisfactory. The scrutinising has been applied to hard and soft bit error sequences in addition to packet error sequences.

The DPBGM can outperform the proposed models. However, it is not desirable in the industry as it has a deficiency in the process of generating error bursts. This deficiency can easily be tackled by changing the generation process. Therefore, a new model related to the DPBGM has been suggested in Chapter 4 which can eliminate the flaw so that the model becomes more desirable.

Chapter 4

Deterministic Process based Generative Models at Packet Level

4.1 Introduction

Most of the generative models in the literature are designed at bit level, but their parameters can easily be tuned to adapt the packet level. The authors of [83–87] addressed the packet-level error modelling.

The DPBGM performance has shown superiority over the other discussed generative models by fitting the descriptive model. Therefore, it receives more attention in this chapter. The DPBGM has been developed for bit-level error sequences and its performance has been shown in [70]. Although some of the DPBGM's performance for packet level have been shown in Subsection 3.2.3.1, we show here more performance results for it. In addition, the packet DPBGM parametrisation is demonstrated.

We also provide a modification procedure to the DPBGM in order to mitigate its disadvantage of borrowing error bursts from the reference error sequences in the stage of generating new error sequences. This behaviour is very undesirable in error modelling.

4.2 Packet-Level DPBGM

The utilised deterministic fading process $\tilde{\zeta}(t)$ of (2.2) is sampled with a reliable sampling period T_A. This is natural when considering block or packet transmissions, especially when the packet is short and the data rate is high, i.e., data rate is much greater than Doppler frequency. In this case, it is reasonable to assume that the various bits of a same packet experience approximately the same channel conditions [18].

For this purpose, a threshold detector with a chosen threshold value r_{th} is then applied to the sampled deterministic process $\tilde{\zeta}(kT_A)$, where k is a non-negative integer. During the simulation, the level of the deterministic process fluctuates and crosses the given threshold r_{th} along the time axis. If the level of $\tilde{\zeta}(kT_A)$ is larger than r_{th}, the model's output produces error-free sample, whereas error sample occurs when the level of $\tilde{\zeta}(kT_A)$ is less than r_{th}. The counts of consecutive error samples or error-free samples, in the corresponding fading and inter-fading intervals of $\tilde{\zeta}(t)$, are the lengths of the error bursts and error-free bursts. This is the mechanism for obtaining the error burst length generator $\widetilde{\mathbf{EB}}_{rec}$ and error-free burst length generator $\widetilde{\mathbf{EFB}}_{rec}$. Note that any symbol that has a tilde ($\tilde{}$) sign is related to the generative model.

For the parametrisation purpose, the level-crossing rate (LCR) $\tilde{N}_{\zeta}(r_{th})$ at the chosen threshold r_{th} is fitted to the desired occurrence rate $R_{EB} = N_p \mathcal{N}_{EB}/T_t$ of error bursts. Here, N_p stands for the packet size, $\mathcal{N}_{EB}$ is the total number of error bursts, and T_t denotes the total transmission time of the reference transmission system, from which the reference packet error sequence of length N_t bits is obtained. The ratio $\tilde{\mathcal{R}}_B$ of the ADF $\tilde{T}_{\zeta_-}(r_{th})$ at r_{th} to the average duration of inter-fades (ADIF) $\tilde{T}_{\zeta_+}(r_{th})$ at r_{th} is approximated to the desired ratio $\mathcal{R}_B = M_{EB}/M_{EFB}$, where M_{EB} and M_{EFB} are the mean values of the error burst and error-free burst lengths, respectively, of the reference packet error sequence. Moreover, the sampling interval T_A is chosen carefully, as specified below, in order to detect most of the level crossings and fading intervals at deep levels.

In (2.4), when using the method of exact Doppler spread (MEDS) with $N_i \geq 7$, the LCR $\tilde{N}_{\zeta}(r_{th})$ of $\tilde{\zeta}(t)$ approximately fits the LCR $N_{\zeta}(r_{th})$ of a Rayleigh process.

Moreover, the ADF $\tilde{T}_{\zeta_-}(r_{th})$ and the ADIF $\tilde{T}_{\zeta_+}(r_{th})$ of $\tilde{\zeta}(t)$ approximate very well the corresponding quantities of a Rayleigh process $T_{\zeta_-}(r_{th}) = \sqrt{\frac{2\pi}{\beta}} \frac{\sigma_0^2}{r_{th}} \left[\exp(\frac{r_{th}^2}{2\sigma_0^2}) - 1\right]$ and $T_{\zeta_+}(r_{th}) = \sqrt{\frac{2\pi}{\beta}} \frac{\sigma_0^2}{r_{th}}$, respectively. Therefore, $\tilde{\mathcal{R}}_B$ can be approximated as

$$\tilde{\mathcal{R}}_B \approx \frac{T_{\zeta_-}(r_{th})}{T_{\zeta_+}(r_{th})} = \exp(\frac{r_{th}^2}{2\sigma_0^2}) - 1 \; . \tag{4.1}$$

As observed, the second order statistics of $\tilde{\zeta}(kT_A)$ are determined by the parameter vector $\mathbf{\Psi} = (N_1, N_2, r_{th}, \sigma_0, f_{max}, T_A)$. The parameter r_{th} can be assigned such that it is much less than 1. In order to find other parameters, we apply $\mathcal{R}_B = \tilde{\mathcal{R}}_B$ and $R_{EB} = N_\zeta(r_{th})$ to get $f_{max} = \frac{N_p \mathcal{N}_{EB}(1+\mathcal{R}_B)}{T_t\sqrt{2\pi \ln(1+\mathcal{R}_B)}}$. $T_A \approx \frac{4\sigma_0[\exp(\frac{r_{th}^2}{2\sigma_0^2})-1]}{\sqrt{5}\pi r_{th} f_{max}} \sqrt{-1 + \sqrt{1+10q_s/3}}$, where q_s is a very small value, e.g., $q_s = 0.01$. This value determines the maximum measurement error of the LCR. This implies that the probability of undetectable level crossings at r_{th} is not larger than q_s [67, 88].

Finally, the sampled deterministic process $\tilde{\zeta}(kT_A)$ can be simulated within the necessary time interval $[0, \tilde{T}_t]$. Here, $\tilde{T}_t = T_t \tilde{N}_t / N_t$ with $\tilde{N}_t$ denoting the required length of the generated packet error sequence. The total numbers of the generated error bursts $\tilde{\mathcal{N}}_{EB}$ and error-free bursts $\tilde{\mathcal{N}}_{EFB}$ can approximately be estimated from $\tilde{\mathcal{N}}_{EB} = \lfloor \frac{\tilde{N}_t}{N_t} \mathcal{N}_{EB} \rfloor$ and $\tilde{\mathcal{N}}_{EFB} = \lfloor \frac{\tilde{N}_t}{N_t} \mathcal{N}_{EFB} \rfloor$, respectively. Here, $\lfloor x \rfloor$ stands for the floor of x. Consequently, an error burst length generator $\widetilde{\mathbf{EB}}_{rec}$ with $\tilde{\mathcal{N}}_{EB}$ entries and an error-free burst length generator $\widetilde{\mathbf{EFB}}_{rec}$ with $\tilde{\mathcal{N}}_{EFB}$ entries are derived.

The mappers of (2.7) and (2.8) are then applied as in Chapter 2. The mapping process yields the correct $\widetilde{\mathbf{EB}}_{rec}$ and $\widetilde{\mathbf{EFB}}_{rec}$, from which the corresponding error burst and error-free bursts, respectively can be produced. Since error bursts consist of clusters and gaps combined in sequence, it is convenient to create parameter vectors $\widetilde{\mathbf{ECG}}_j$ ($j = 1, 2, \ldots, \tilde{\mathcal{N}}_{EB}$), which reflect the construction of each error burst from $\widetilde{\mathbf{EB}}_{rec}$ as error cluster and gap lengths. Therefore, all vectors $\mathbf{ECG}_i$ corresponding to error bursts with length m_e are found in $\mathbf{EB}_{rec}$. Thereafter, for all error bursts with the same length m_e in $\widetilde{\mathbf{EB}}_{rec}$, random $\widetilde{\mathbf{ECG}}_j$ are allocated from all possible vectors $\mathbf{ECG}_i$. This is the procedure for generating the error bursts. Error-free bursts, on the other hand, consist of zeros only. Therefore, they are obtained by generating a series of

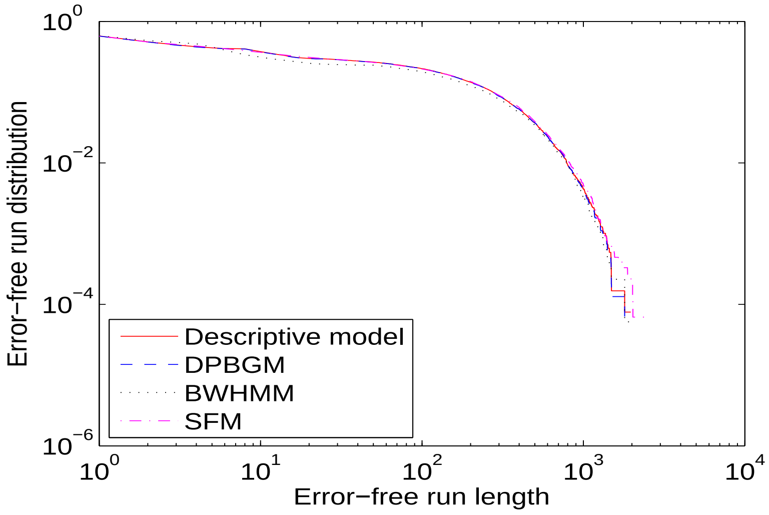

FIGURE 4.1: EFRDs of the descriptive model obtained from the EGPRS system and three generative models.

zeros for each length in $\widetilde{\mathbf{EFB}}_{rec}$. By combining generated error bursts with error-free bursts in succession, an entire packet error sequence is constructed.

For the above three design steps of the DPBGM, the first two steps (parametrisation and mapping) are called the simulation set-up phase, while the last step (generation of packet error sequences) is called the simulation run phase.

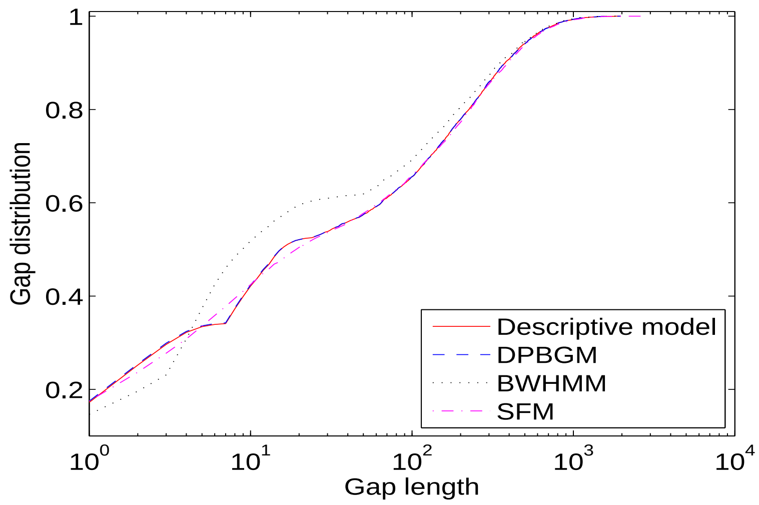

FIGURE 4.2: GDs of the descriptive model obtained from the EGPRS system and three generative models.

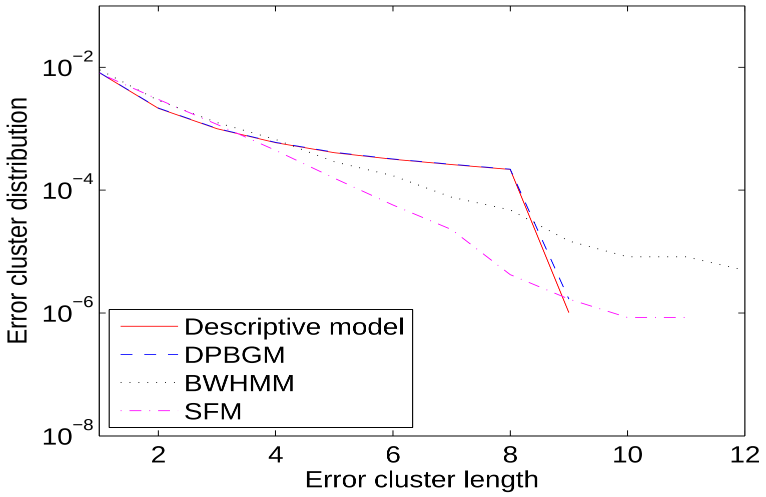

FIGURE 4.3: ECDs of the descriptive model obtained from the EGPRS system and three generative models.

4.2.1 Simulation Results and Discussions

The performance criteria are normally evaluated by working out the burst error statistics. One generative model is preferred over others if its burst error statistics fit very well those of the reference packet error sequences obtained directly from the EGPRS

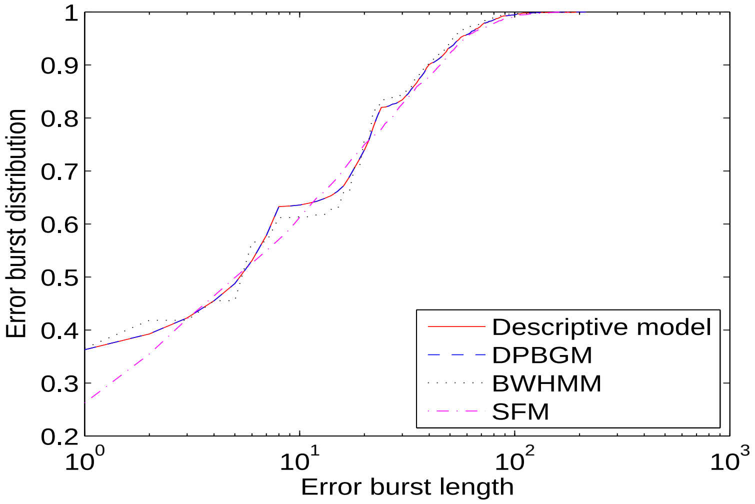

FIGURE 4.4: EBDs of the descriptive model obtained from the EGPRS system and three generative models.

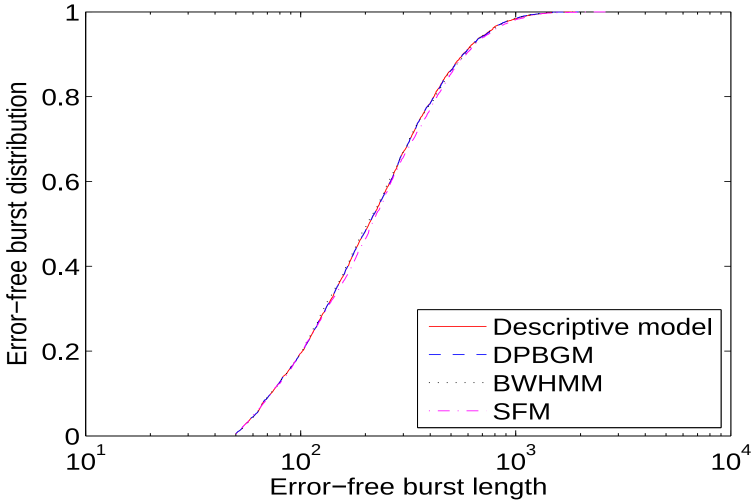

Figure 4.5: EFBDs distributions of the descriptive model obtained from the EG-PRS system and three generative models.

system. Especially, the most important burst error statistics such as the BEPD which is helpful for designing the high layer protocols.

Here, we use the same descriptive models, DPBGM, and SFM as the one of Subsection 3.2.3.1. The value of $N_p = 456$ for the DPBGM. We additionally add here the modelling of BWHMM for further comparison demonstration. The simulation set-up

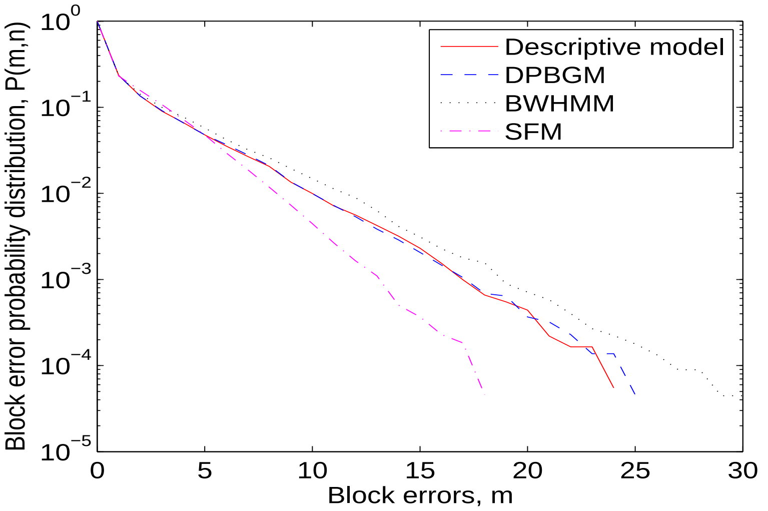

Figure 4.6: BEPDs of the descriptive model obtained from the EGPRS system and three generative models.

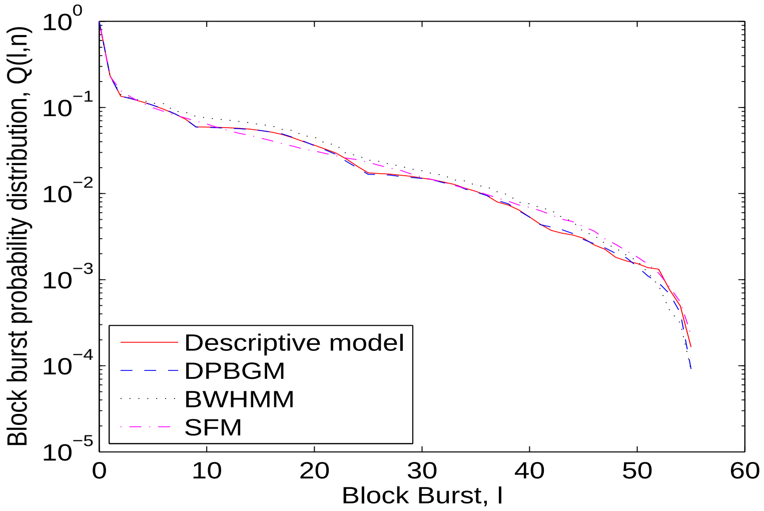

Figure 4.7: BBPDs distributions of the descriptive model obtained from the EGPRS system and three generative models.

phase of the BWHMM involves extracting the error bursts from the error sequence, that are then divided into smaller blocks of length $L = 2$. Based on the maximum number of errors in L, the error bursts are classified. In our simulations, the number of classes is 3 and the total number of states is 100.

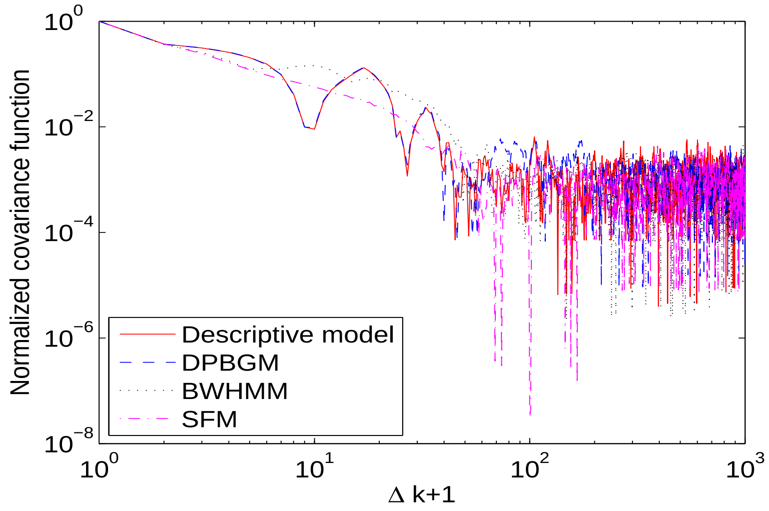

Figure 4.8: NCFs of the descriptive model obtained from the EGPRS system and three generative models.

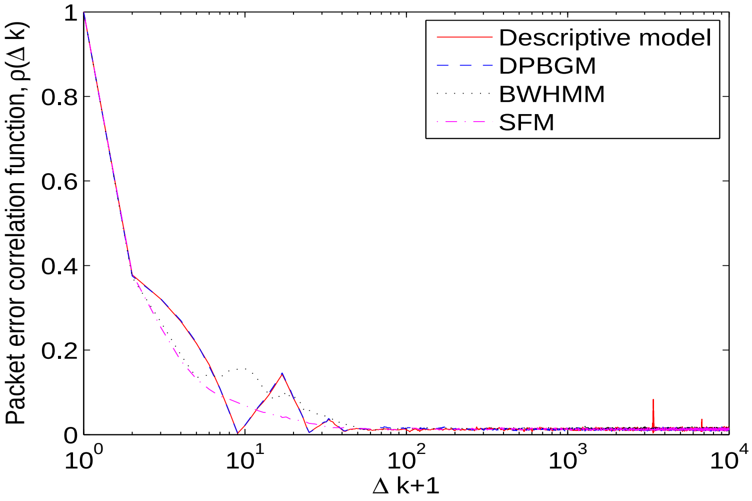

FIGURE 4.9: PECFs distributions of the descriptive model obtained from the EGPRS system and three generative models.

FIGURES 4.1–4.12 depict the behaviour of the descriptive model and the generative models of the DPBGM, BWHMM, and SFM in terms of EFRDs, GDs, ECDs, EBDs, EFBDs, BEPDs, BBPDs, NCs, PECFs, GCFs, MGDs against the multigap length of order 10, and MGDs against the multigap length of order 100, respectively. In general, the DPBGM outperforms the SFM and BWHMM in terms of

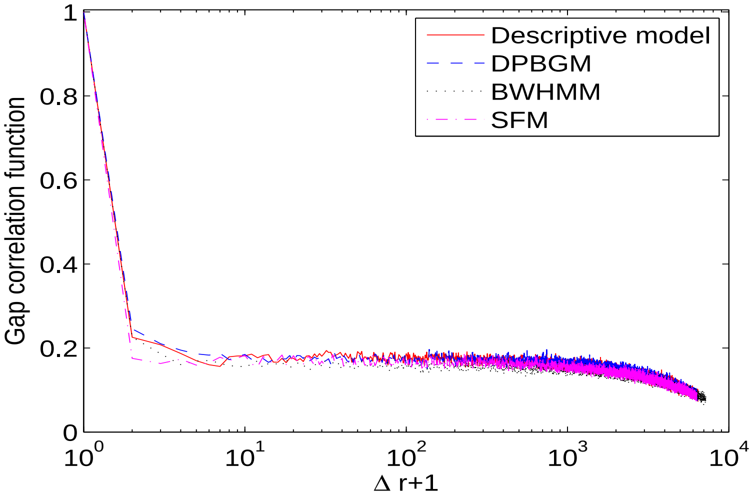

FIGURE 4.10: GCFs of the descriptive model obtained from the EGPRS system and three generative models.

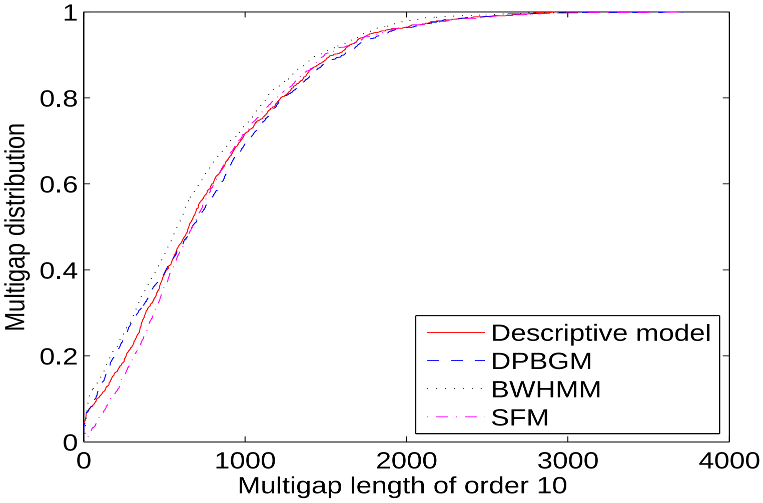

FIGURE 4.11: MGDs with order 10 of the descriptive model obtained from the EGPRS system and three generative models.

accuracy by well fitting the descriptive model for all the defined burst error statistics. An exception is the MGD against the multigap length of order 100 (FIGURE 4.12), for which the SFM behaves the best. However, the SFM fails to describe some of the desired burst error statistics, as can be seen from the large deviations for ECD (FIGURE 4.3), EBD (FIGURE 4.4), BEPD (FIGURE 4.6), NCF (FIGURE 4.8),

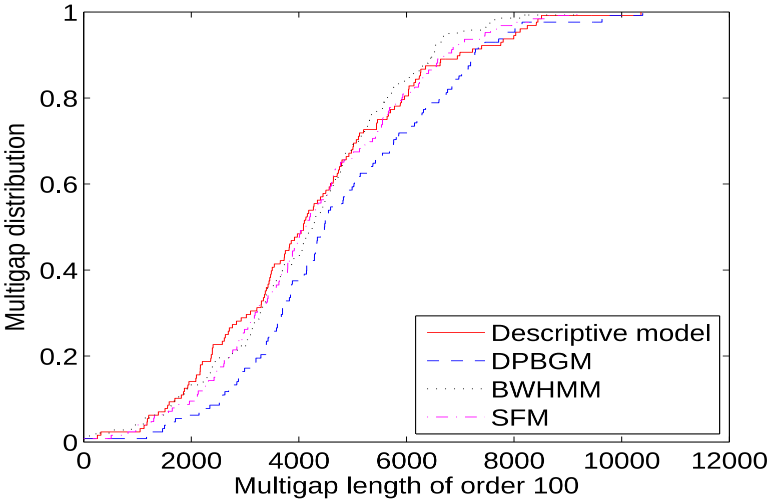

FIGURE 4.12: MGDs with order 100 of the descriptive model obtained from the EGPRS system and three generative models.

and PECF (Figure 4.9), whereas it satisfactorily matches the descriptive model for EFRD (Figure 4.1), GD (Figure 4.2), EFBD (Figure 4.5), BBPD (Figure 4.7), and MGDs (Figures 4.11 and 4.12). The BWHMM fails to characterise the GD (Figure 4.2), ECD (Figure 4.3), BEPD (Figure 4.6), NC (Figure 4.8), and PECF (Figure 4.9) adequately, while the performance is acceptable for the rest burst error statistics. This is because the BWHMM is best designed to be used for error sequences having error bursts with bell-shaped error density, while this property is difficult to be found in packet error sequences. Nonetheless, the BWHMM shows better performance than the SFM in terms of ECD (Figure 4.3), EBD (Figure 4.4), EFBD (Figure 4.5), BEPD (Figure 4.6), and PECF (Figure 4.9), whereas it is not better than the SFM for the remaining burst error statistics.

In terms of model complexity and simulation efficiency, all the three generative models require two phases: simulation set up phase and simulation run phase. For the set up phase, all the three models have high complexity and require a long simulation time, which depends on individual experiences and are difficult to compare. For the simulation run phase, the DPBGM has the minimum complexity followed by the BWHMM and then the SFM. Using a PC with a 2.4 GHz processor, the DPBGM, BWHMM, and SFM need 0.422, 1.125, and 3.422 s, respectively. Thus, the DPBGM outperforms the BWHMM and SFM in terms of accuracy as well as efficiency.

4.3 Deterministic Process based HMM

It is well explained in the former Sections that the DPBGMs do not create new error bursts by its own in the design step of generating error sequences. The DPBGMs only create the lengths of error bursts and error-free bursts. Then, they retrieve error bursts of the same length directly from the reference error sequences. This behaviour restricts the capability of DPBGMs to adaptively generate new error sequences at different channel conditions. Also, if only reference burst error statistics are given while target error sequences are not available, the DPBGM is of no use but other generative models (e.g., SFM and BWHMM) can still generate error sequences.

To overcome the drawback of DPBGMs, we have proposed a new packet-level generative model that replaces the problematic last design step (generation of error sequences) of the DPBGMs by using a HMM. This enhances the adaptability of the new model to generate error bursts and consequently error sequences at various channel conditions. We call the new model the deterministic process based HMM (DPB-HMM). Note that the proposed DPB-HMM can also be used for generating bit-level error sequences. The chosen HMM can easily be parametrised based on the procedure developed in [51]. Unlike the model in [51], the DPB-HMM keeps the link between the fading process and the underlying Markov process. Furthermore, the proposed DPB-HMM is less complicated than the BWHMM.

Therefore, we develop a general design procedure for generating packet-level error sequences using a properly parametrised and sampled deterministic fading process, followed by a threshold detector and two parallel mappers, combined with a HMM. The first two design steps, i.e., parametrisation of a sampled deterministic fading process with a threshold detector and mapping process, of the proposed DPB-HMM are called 'simulation set-up phase', which is very similar to that of a DPBGM. The major challenge for the DPB-HMM is the third design step, i.e., generating packet-level error sequences using a HMM without retrieving error bursts from reference error sequences as in DPBGM. We call the third design step the 'simulation run phase'. FIGURE 4.13 illustrates the structure of the proposed DPB-HMM. Two embedded classifiers or layers are added after the two mappers.

The first layer is to generate error bursts and error-free bursts, while the second layer is to construct error bursts in more detail, which include error clusters and gaps.

In the first layer, the generation of error-free bursts (represented by the **0** state or error-free state) is straightforward since each entry of $\widetilde{\mathbf{EFB}}_{rec}$ is basically presented as the number of consecutive zeros. The error bursts are more complicated than error-free bursts, because they consist of clusters (ones) and gaps (zeros) of different lengths. A good practice of generating error bursts is to divide the modified $\widetilde{\mathbf{EB}}_{rec}$ into different classes to simplify their generation. Each class, representing an error-state in the first-layer Markov chain, conveys a common structural behaviour of error bursts. To balance the simplicity and accuracy of the HMM, the number of classes or error-states should be properly chosen. Since we know $\mathbf{EB}_{rec}$ from the descriptive model, we can count the number of errors in each length. Consequently, we can use the ratio of the number of errors to the error burst length (or local error density) as a criterion to divide the error bursts into classes. This is reasonable because our aim is to characterise the errors in the error bursts. Therefore, the range of the resulting ratios is divided into N states, i.e., $S = \{s_0, s_1, ..., s_N\}$, where S is the set of error

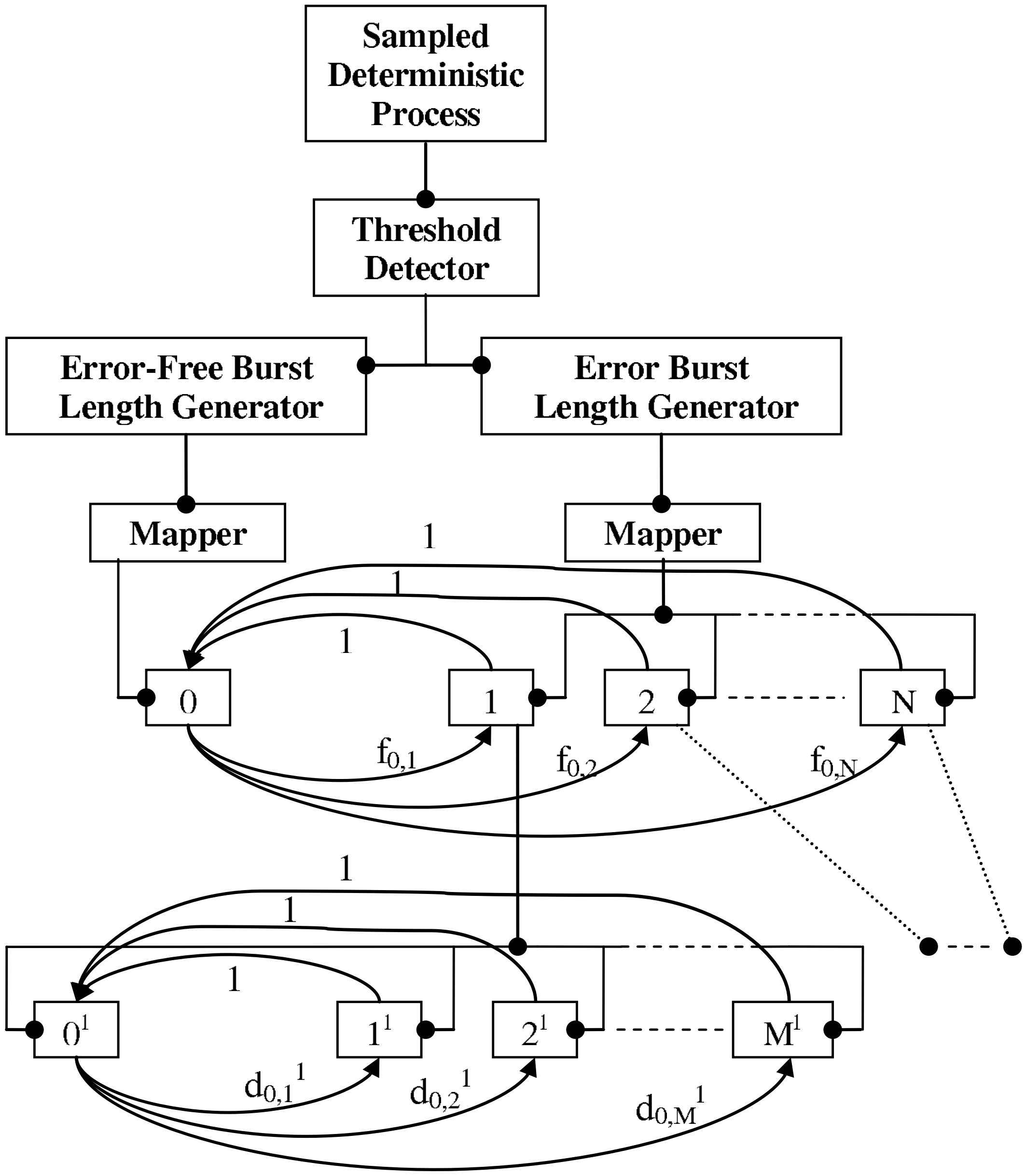
Sampled Deterministic Process
Threshold Detector
Error-Free Burst Length Generator
Error Burst Length Generator
Mapper
Mapper
1
1
1
0
1
2
N
f0,1
f0,2
f0,N
1
1
1
0^1
1^1
2^1
M^1
d0,1^1
d0,2^1
d0,M^1

FIGURE 4.13: The DPB-HMM.

burst states. Each state is allocated an approximately equal number (e.g., 300) of error bursts corresponding to the divided range of ratios.

Once the error bursts are classified and the number of error states is defined, we need to determine the state transition matrix $\mathbf{F} = (f_{i,j})$, $0 \leq i, j \leq N$. The element $f_{i,j}$ is the transition probability from the current state $Q_t = s_i$ at time t to the next state $Q_{t+1} = s_j$ at time $t+1$. From the first layer of the hidden Markov process in FIGURE 4.13, it is clear that the transition from any error state to the error-free state is compulsory so that we can generate the required packet error sequence in a successive manner of error bursts and error-free bursts. This means that the transition probability $f_{i,j} = 1$ for $1 \leq i \leq N$ and $j = 0$. Moreover, there is no self-transition in any state, i.e., $f_{i,j} = 0$ for $i = j$. In addition, there is no transition between the different error burst states, i.e., $f_{i,j} = 0$ for $1 \leq i, j \leq N$. The state transition from the error-free state to any of the error burst states depends mainly on the number of error bursts in each error burst state. Based on the above analysis, the transition probability $f_{i,j}$ can be expressed by

$$
\begin{aligned}
f_{i,j} &= P\left[Q_{t+1} = s_j \,|\, Q_t = s_i\right], 0 \leq i, j \leq N \\
&= \begin{cases} 1, & 1 \leq i \leq N,\ j = 0 \\ \frac{N_{EB,j}}{\sum_{j=1}^{N} N_{EB,j}} \approx \frac{N_{EB,j}}{N \times N_{EB,N}}, & i = 0,\ 1 \leq j \leq N \\ 0, & \text{otherwise} \end{cases}
\end{aligned}
\tag{4.2}
$$

where $N_{EB,j}$ denotes the number of error bursts in State s_j. It follows that the state transition matrix is given by

$$
\mathbf{F} = \begin{pmatrix} 0 & f_{0,1} & \cdots & f_{0,N} \\ 1 & 0 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 0 & \cdots & 0 \end{pmatrix}. \tag{4.3}
$$

Each error burst can be further divided into error clusters and gaps (with lengths less than η). Therefore, the second layer of the hidden Markov process, emitting from each error state in the first layer, is used to construct the error bursts in detail. For this purpose, we need to find the error cluster recorder $\mathbf{EC}_{rec}(s_j)$ and gap recorder $\mathbf{G}_{rec}(s_j)$ for each error burst state of the first layer. For the second layer, we assign only one substate for the gaps, and a number of substates for the error clusters. The number of error cluster substates depends on the number of lengths the error clusters has in each state. Each error cluster length has a substate. The number of error cluster substates is denoted by M, i.e., $V_{O_u} = \{v_{0_u}, v_{1_u}, ..., v_{M_u}\}$, where V_{O_u} is the set of substates in state S_u ($u = j$ when $j \neq 0$). Here, $O = 0, 1, ..., M$. The elements $(d^u_{h,k})$ of the substate transition matrix $\mathbf{D}_u = (d^u_{h,k})$ can be calculated as

$$d^u_{h,k} = P\left[R_{t+1} = v_{k_u} | R_t = v_{h_u}\right], 0 \leq h_u, k_u \leq M_u$$

$$= \begin{cases} 1, & 1 \leq h_u \leq M_u,\ k_u = 0 \\ \frac{N_{EC,k_u}}{\sum_{k_u=1}^{M_u} N_{EC,k_u}}, & h_u = 0, 1 \leq k_u \leq M_u \\ 0, & \text{otherwise} \end{cases} \tag{4.4}$$

where $d^u_{h,k}$ is the transition probability from Substate v_{h_u} to Substate v_{k_u}, R_t is the current substate at time t, and N_{EC,k_u} is the number of error clusters in v_{k_u}. It is apparent that the substate transitions depend mainly on N_{EC,k_u}.

In the process of generating error bursts in the hidden second layer, we must make sure that the generation starts and ends with error clusters (based on the definition of an error burst). Therefore we define the initial substate distribution vector $\mathbf{\Pi}_u = (\pi^u_k)$ as

$$\mathbf{\Pi}_u = (0, d_{0,1}, ..., d_{0,M_u}) \tag{4.5}$$

where π^u_k is the probability of v_{k_u} to be an initial or final substate. The generation of gap lengths in each group depends on their length probability distribution.

Once the states and state transition matrices in the two layers of the hidden Markov process are determined, packet error sequences with consecutive error bursts (with error clusters and gaps) and error-free bursts can be generated.

To recap, we obtain the consecutive error burst lengths $\widetilde{\mathbf{EB}}_{rec}$ and error-free burst lengths $\widetilde{\mathbf{EFB}}_{rec}$ from the deterministic fading process. Basically, the length of the error-free burst produces an error-free burst by interpreting the length to the number of zeros. Whereas, the error burst length determines in which state the error burst shall be constructed. If one error-burst length is located in many different states, we randomly (uniformly distributed) choose one of the states. Finally, according to the error burst length, an error burst is created in the hidden second layer. The generated error burst length does not necessarily have the exact length as required, because the gaps are chosen randomly and the final error burst length could exceed the required length. The difference between the required and generated error burst lengths should be small (e.g., less than 10 bits). Otherwise, the current error burst should be discarded and the generation process should be repeated. The generation of a packet error sequence is stopped once its length reaches $\tilde{N}_t$ or exceeds it. At the end, the percentage of the generated error packets among all the generated packets from each state must approximately match that of the corresponding state of the reference error sequence. If the difference is large, it means that the error bursts of this state cannot convey the structural behaviour of those of the reference sequence. In this case, this state should be divided initially into two parts. Each part should be tested again for the same purpose. This might happen at good channel conditions when the errors are less correlated.

4.3.1 Simulation Results and Discussions

Again, we adopt the widely used SFM [22], BWHMM [45], and DPBGM [70] to show their burst error statistics against those of the descriptive model and the proposed DPB-HMM. The descriptive model of the EGPRS has been illustrated in the previous sections as well as the DPBGM, SFM, and BWHMM. However, the CIR value here is 8 dB. For the DPBGM, $\boldsymbol{\Psi} = (9, 10, 0.09, 0.1001, 2.32kHz, 1.432ns)$ with $\eta = 50$, $\mathcal{R}_B = 0.4979$, and $\mathcal{N}_{EB} = 6044$ hold. In case of the BWHMM, the number of classes is 9, the total number of states is 321, and $L = 8$.

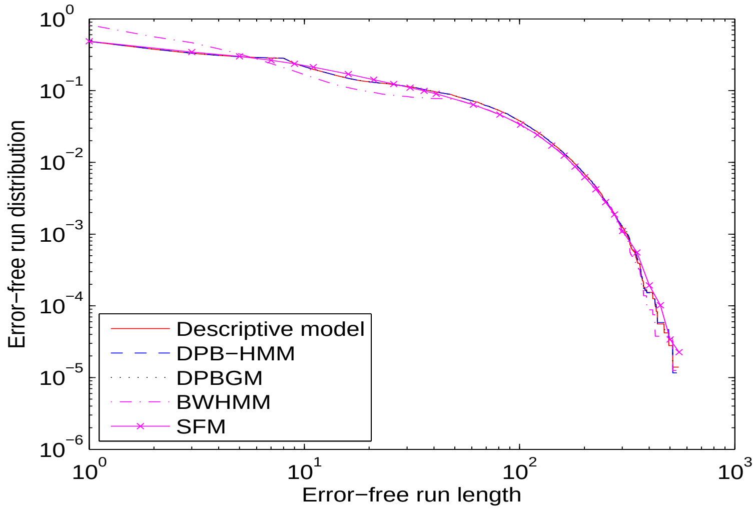

FIGURE 4.14: EFRDs of the descriptive model obtained from the EGPRS system and different generative models.

FIGURES 4.14–4.22 show the burst error statistics of the descriptive model and the four generative models (DPB-HMM, DPBGM, BWHMM, and SFM) in terms of their EFRDs, GDs, ECDs, EBDs, EFBDs, BEPDs, BBPDs, NCFs, and PECFs, respectively. For NCFs (FIGURE 4.21), the behaviour of different generative models becomes crowded at the end. Therefore, we consider the comparison up to a value where the

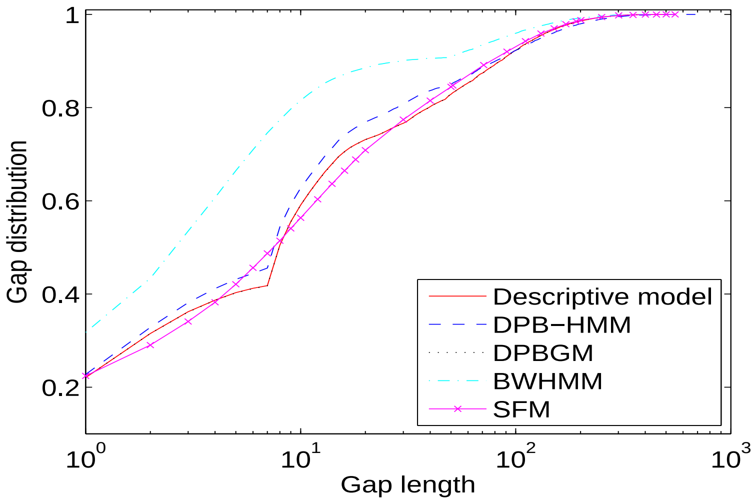

FIGURE 4.15: GDs of the descriptive model obtained from the EGPRS system and different generative models.

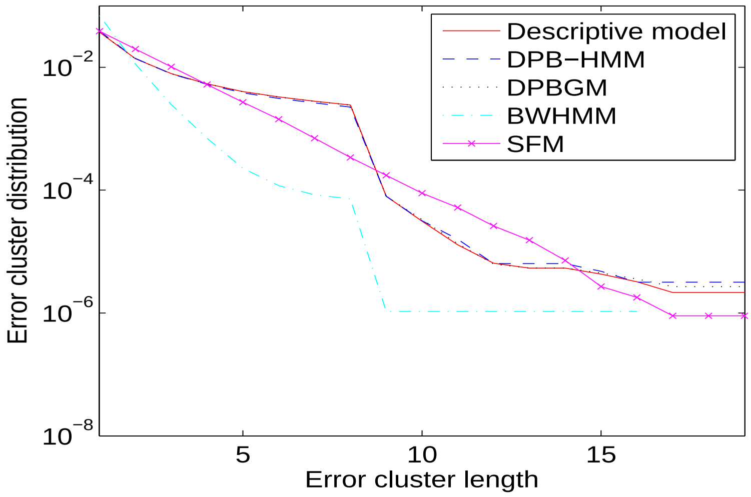

FIGURE 4.16: ECDs of the descriptive model obtained from the EGPRS system and different generative models.

covariance is greater than 0.01. That value determines the length of the used interleaver in the system design. The interleaver converts bursty errors to random errors to be handled by error control schemes.

Regarding the generative models performance, the DPBGM generally shows the best behaviour among all the four generative models regarding the accuracy of matching

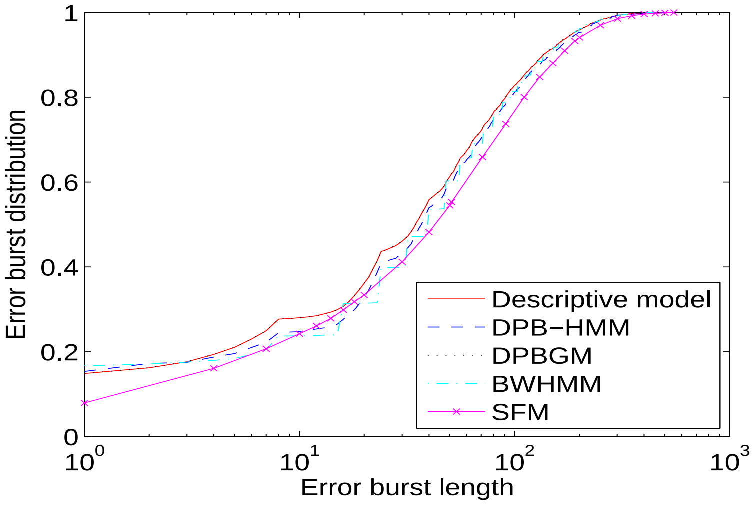

FIGURE 4.17: EBDs of the descriptive model obtained from the EGPRS system and different generative models.

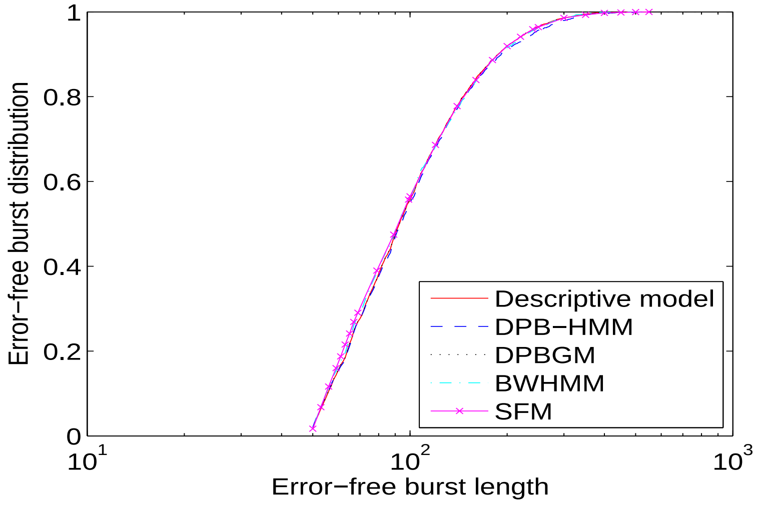

FIGURE 4.18: EFBDs of the descriptive model obtained from the EGPRS system and different generative models.

the descriptive model for all the defined burst error statistics. However, the DP-BGM retrieves the error bursts from the reference packet error sequence rather than constructing them by itself, which explains its good performance. Nonetheless, this process of borrowing error bursts from the reference error sequence limits its adaptability to generate new error sequences with different channel conditions using the

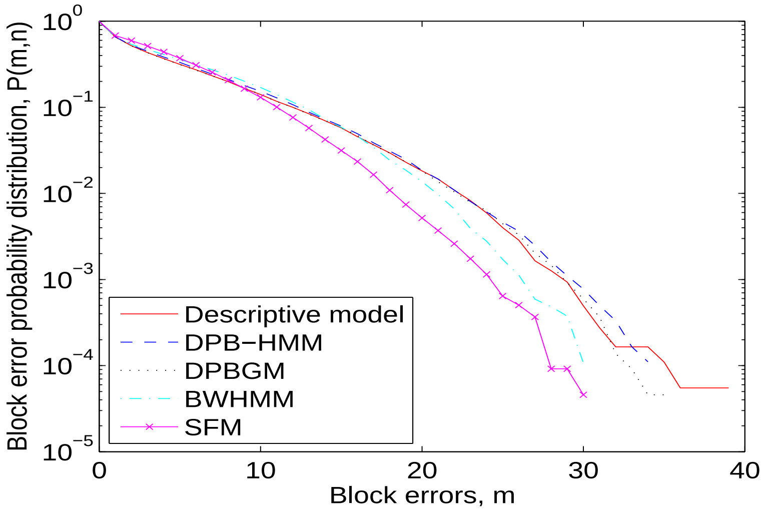

FIGURE 4.19: BEPDs of the descriptive model obtained from the EGPRS system and different generative models.

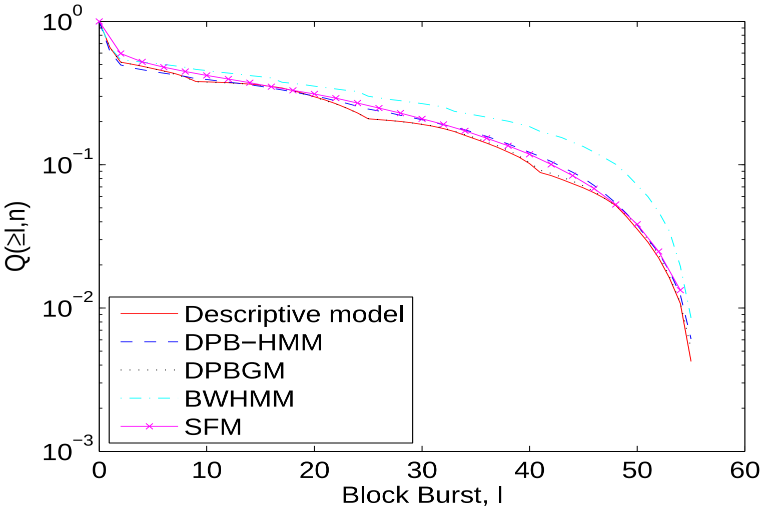

FIGURE 4.20: BBPDs of the descriptive model obtained from the EGPRS system and different generative models.

existing error sequences and is therefore not preferred in generative models. The SFM, BWHMM, and the proposed DPB-HMM all create error bursts directly after they are parametrised.

The SFM can approximate the desired EFRD, GD, EFBD, and BBPD (see FIGURES 4.14, 4.15, 4.18, and 4.20, respectively) with a satisfactory performance but

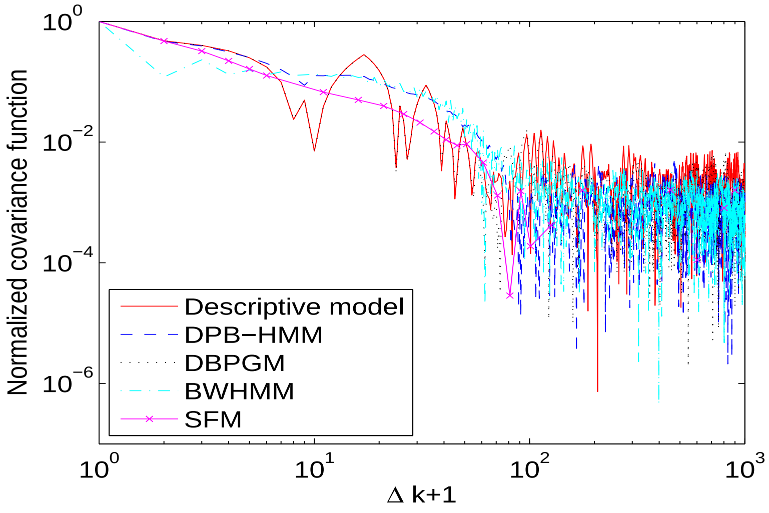

FIGURE 4.21: NCFs of the descriptive model obtained from the EGPRS system and different generative models.

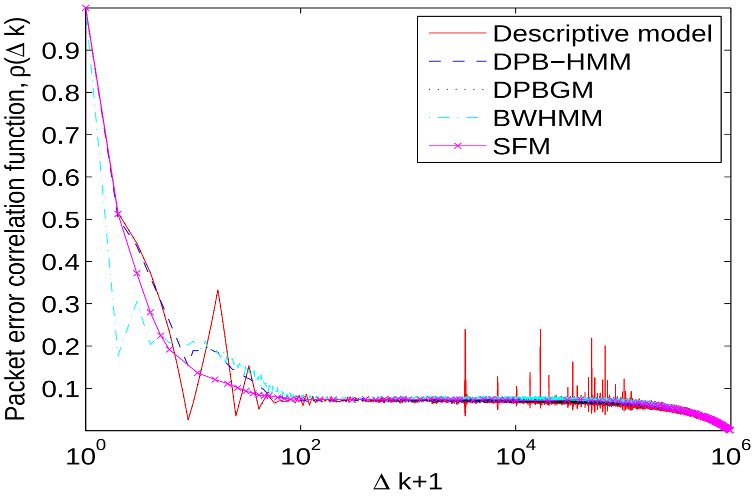

FIGURE 4.22: PECFs of the descriptive model obtained from the EGPRS system and different generative models.

fail to describe the desired ECD, EBD, BEPD, NCF, and PECF (see FIGURES 4.17, 4.19, 4.21, and 4.22, respectively) with a large deviation. The BWHMM can only match well the desired EBD and EFBD (see FIGURES 4.17, and 4.18, respectively) while fail to characterise the rest burst error statistics. This is because the BWHMM [45] is best designed to be used for error sequences having error bursts with a bell-shaped error density, which is not a property for our reference packet error sequences. The SFM outperforms the BWHMM with regard to the EFRD, GD, ECD, BBPD, NCF, and PECF (see FIGURES 4.14, 4.15, 4.16, 4.20, 4.21, and 4.22, respectively), whereas the BWHMM outperforms the SFM in terms of EBD, EFBD, and BEPD (see FIGURES 4.17–4.19, respectively).

The proposed DPB-HMM can approximate most of the desired burst error statistics very well, including the EFRD, ECD, EFBD, BEPD, BBPD, and NCF (see FIGURES 4.14, 4.16, 4.19–4.21, respectively). The accuracy of matching the desired GD, EBD, and PECF (see FIGURES 4.15, 4.17, and 4.22, respectively) using the DPB-HMM is also acceptable but not as good as that using the DPBGM. Note that the BEPD, BBPD, NCF, and PECF are the four most important ones amongst all burst error statistics. When the users choose the generative models, they should pick up the one which has the best fit to the 4 most important burst error statistics of the descriptive model with proper consideration to the model complexity and adaptability. It is also worth mentioning that using generative models can significantly save computation time compared with using the descriptive model. The descriptive model needs approximately 28 minutes for generating a packet error sequence that has a length of 1 million. For generating a packet error sequence with a length of 1.5 million, the DPBHMM, DPBGM, BWHMM, and SFM need for their simulation run phase about 1.1, 0.5, 6.6, and 9.8 seconds, respectively, using a PC with an Intel processor of 2 GHz.

Note that the EBD of the DPB-HMM has a small mismatch from the descriptive EBD. This is because the generated lengths sometimes exceed the required length by a small value. Moreover, the GD of the DPB-HMM does not closely match that of the descriptive model due to some non-generated gaps, but the distribution still mimics the reference one. The DPB-HMM does not fit the distinct breakpoints of the PECF, however, it fits the other parts of the PECF very well.

In summary, the DPB-HMM has a slightly reduced performance compared to the DPBGM but outperforms the SFM and BWHMM. Also, the DPB-HMM eliminates the drawback of the DPBGM with better adaptability and has less complexity compared with the BWHMM. Therefore, the proposed DPB-HMM is deemed to be the best generative model among the four models in terms of the tradeoff between accuracy, complexity, and adaptability.

Please note that we have also tried error sequences with different channels and different CIRs, rather than TU3 IFH with a CIR of 8dB. In all the cases, the burst error statistics of the proposed DPB-HMM can match well those of the descriptive model. The DPB-HMM shows superior performance than SFM and BWHMM. Therefore, the above conclusions remain the same for target error sequences with different channels and different CIRs.

4.4 Summary

We have established a general procedure for developing a fast binary packet-level generative model with a properly parametrised and sampled deterministic process followed by a threshold detector and two parallel mappers. It has been demonstrated that in general the DPBGM exhibits excellent conformity with the descriptive model especially for the most important burst error statistics like the PECF and BEPD which are used in the design and performance evaluation of the MAC layer, link control layer, and high layer wireless communication protocols. The SFM and BWHMM fail to describe most of the burst error statistics and notably the important ones. The SFM outperforms the BWHMM in terms of the EFRD and MGD but is worse than the DPBGM for the same statistics. The BWHMM performs better than the SFM in terms of ECD, EBD, EFBD, BEPD, and PECF, but not better than the DPBGM. The DPBGM has shown its superiority in terms of efficiency as well. However, its method of generating error bursts is not desirable and limits its applications. Therefore, the DPB-HMM has been suggested to tackle the DPBGM's shortcoming.

This shortcoming has been eliminated in the DPB-HMM by replacing the DPBGM's generation process with a simple HMM process. The DPB-HMM outperforms the SFM and BWHMM and has a slightly reduced accuracy performance compared to the DPBGM. In terms of the overall tradeoff between the model accuracy, complexity, and adaptability, the proposed DPB-HMM is deemed as the best amongst all the existing generative models.

Chapter 5

Adaptive Generative Models

5.1 Introduction

All the aforementioned traditional generative models (MMs, HMMs, SCFGs, CAGMs, DPBGMs) were developed for error sequences corresponding to one digital channel with fixed system parameters and channel conditions, e.g., signal-to-noise-ratio (SNR) values. If SNRs change, the whole communication system will have to be simulated again in order to generate new reference error sequences. Consequently, those traditional generative models will have to be developed again with modified model parameters in order to fit the burst error statistics to those of new reference error sequences. However, for appropriate design and performance evaluation of error control schemes and high layer protocols, we need to test transmission schemes with many different channel conditions or SNRs. In this case, using traditional generative models are still very time consuming, as numerous reference error sequences corresponding to different SNRs have to be first produced by the reference communication system. Therefore, it is highly desirable to develop adaptive generative models that can utilise the limited available error sequences in order to attain new error sequences with any SNRs.

In this chapter, we propose four adaptive generative models based on well-known generative models, i.e., SFM, BWHMM, DPBGM, and DPB-HMM, by adjusting some useful model parameters. Reference error sequences (descriptive models) were obtained from uncoded long term evolution (LTE) systems.

The burst error statistics of newly generated error sequences using four adaptive generative models are compared with those of reference error sequences. Also, the bit error rates (BERs) of coded LTE systems are compared using reference error sequences and new error sequences obtained from four adaptive generative models. It is shown that the proposed adaptive DPBGM and DPB-HMM can provide excellent approximation to the desired burst error statistics of reference error sequences and the desired BER of coded LTE systems. However, the adaptive SFM and adaptive BWHMM fail to do so.

5.2 Adaptive SFM (ASFM)

The probability transition matrix for an N-state SFM is [22]

$$\mathbf{T} = \begin{pmatrix} P_{11} & 0 & 0 & 0 & P_{1N} \\ 0 & P_{22} & 0 & 0 & P_{2N} \\ 0 & 0 & \ddots & 0 & \vdots \\ 0 & 0 & 0 & P_{N-1N-1} & P_{N-1N} \\ P_{N1} & P_{N2} & \cdots & P_{NN-1} & P_{NN} \end{pmatrix} \tag{5.1}$$

where P_{ij} is the the probability of transition from State i to State j $(i, j = 1, ..., N)$. Note that States $1, ..., N$ are error-free states, whereas State N is the error state. The transitions between the error-free states are not allowed, i.e., $P_{ij} = 0$ for $i \neq j$ and $i, j<N$. The elements of $\mathbf{T}$ can be determined from the EFRD of the reference error sequence according to [22]

$$P(0^{m_0}|1) = \sum_{i=1}^{N-1} \frac{P_{Ni}}{P_{ii}} P_{ii}^{m_0}, \qquad m_0 > 0. \tag{5.2}$$

The EFRD can also be approximated by the weighted sum of $N-1$ exponentials given by [22]

$$P(0^{m_0}|1) \approx \sum_{i=1}^{N-1} A_i e^{a_i m_0}. \tag{5.3}$$

The parameters A_i and a_i can be found by using an optimisation method or curve fitting technique to match (5.3) with the EFRD obtained from the reference error sequence. Consequently, the elements of $\mathbf{T}$ in (5.1) are obtained as follows [22]

$$P_{ii} = e^{a_i}, \qquad i = 1, \ldots, N-1 \tag{5.4}$$

$$P_{Ni} = A_i \times P_{ii}, \qquad i = 1, \ldots, N-1 \tag{5.5}$$

$$P_{iN} = 1 - P_{ii}, \qquad i = 1, \ldots, N-1 \tag{5.6}$$

$$P_{NN} = 1 - \sum_{i=1}^{N-1} P_{Ni}. \tag{5.7}$$

Once the above transition probabilities, i.e., all the elements of $\mathbf{T}$, are known, an error sequence with any desirable length can be generated. Therefore, the key to generate new error sequences with any required SNRs from two reference error sequences is to obtain the EFRDs of new error sequences, as can be seen from (5.1)–(5.7)

Table 5.1 summarises the ASFM procedure. Suppose that we have two reference error sequences, e.g., X and Y with two different SNRs in dB, e.g., SNR_X and SNR_Y. Their EFRDs are denoted as $P_X(0^{m_0}|1)$ and $P_Y(0^{m_0}|1)$, respectively.

Table 5.1: The ASFM algorithm.

1. Find EFRD corresponding to error sequence X.
2. Find EFRD corresponding to error sequence Y.
3. Apply (5.8)–(5.10) to get the EFRD corresponding to the required error sequence Z.
4. Find the parameters of (5.3) by curve fitting with the newly obtained EFRD.
5. Work out (5.4)–(5.7).
6. Generate the error sequence Z.

The EFRD $P_Z(0^{m_0}|1)$ of a new error sequence Z with different SNR in dB can be obtained from

$$P_Z(0^{m_0}|1) = [P_X(0^{m_0}|1)]^{\alpha} \times [P_Y(0^{m_0}|1)]^{\beta} \tag{5.8}$$

where the exponential indices α and β are given by

$$\alpha = \left| \frac{SNR_Z - SNR_Y}{SNR_X - SNR_Y} \right| \tag{5.9}$$

and

$$\beta = \left| \frac{SNR_Z - SNR_X}{SNR_X - SNR_Y} \right| \tag{5.10}$$

respectively.

Figure 5.1 shows the EFRDs of some reference error sequences (SNRs=1, . . . , 5 dB) and newly generated error sequences with SNRs=2 dB and 4 dB. Note that the EFRD of the new error sequence with SNR=2 dB (SNR=4 dB) was obtained from the EFRDs of reference error sequences with SNRs=1 dB (3 dB) and 3 dB (5 dB), according

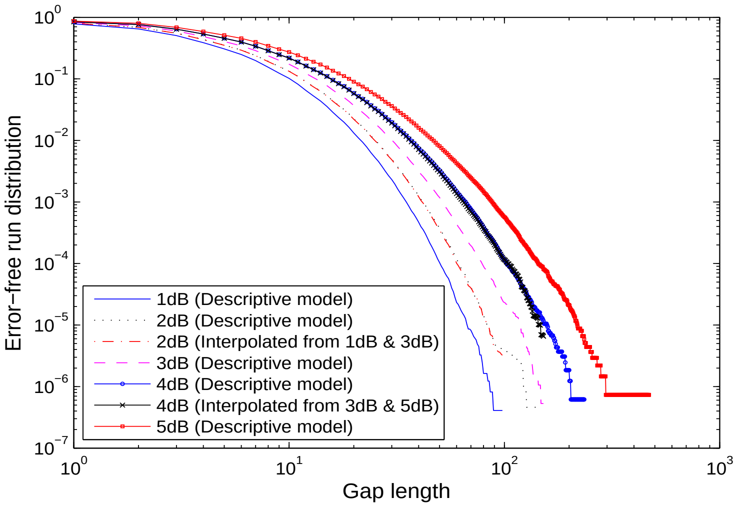

Figure 5.1: EFRDs of the descriptive model obtained from the LTE system at different SNRs.

to (5.8)–(5.10). It is clear that the EFRDs of new error sequences can match those of reference error sequences with the same SNR very well, which verifies the proposed adaptive technique, as shown in (5.8)–(5.10).

5.3 Adaptive Baum-Welch based HMM (ABWHMM)

For the BWHMM, 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 then 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 2 4 1. In this way, the error bursts are converted into a compact format and they then form the vectors $\mathbf{NEL}_h(h = 1, \ldots, \mathcal{N}_{EB})$. The largest number in $\mathbf{NEL}_h$ vector is called the peak number of errors (PNE). Therefore, we have h numbers of PNE. In the previous example, PNE=4 holds.

The next step is to classify the error bursts into N disjoint classes (submodels or bursty states) according to $\xi(N-1)+1 \le \text{PNE} \le \xi N$ where ξ is a positive integer number. Afterwards, the compact blocks of each state are used to train hidden Markov submodels using BW algorithm [47]. Each submodel contains one class of error bursts.

It is already mentioned that in order to build the BWHMM submodels, the parameters N, D, and the set $\lambda = \{A, B, \mathbf{\Pi}\}$ must be specified. The value of N can be decided according to the guidelines in [45]. Given a set of observation sequences (error burst blocks) representing the compact 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 utilised to maximise the probability $\Gamma = \prod_{k=1}^{K} P(\lambda|\mathbf{O}^k)$ [45]. In our previous example, $\mathbf{O} = \{2, \ 4, \ 1\}$.

Once the optimised transition probabilities are found, 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 in the end.

Table 5.2 explains the ABWHMM procedure. To generate new error sequences with any desirable SNRs from two reference error sequences, we should find out the most important feature of the BWHMM. In fact, error models aim to identify the errors and their distribution in the error bursts and this is recognised by the $\mathbf{NEL}_h$ vector. From $\mathbf{NEL}_h$, we can know the number of errors in each block for each error burst. Consequently, from two $\mathbf{NEL}_h$ vectors, i.e., $\mathbf{NEL}_{hX}$ and $\mathbf{NEL}_{hY}$ corresponding to two reference error sequences X and Y, we can obtain $\mathbf{NEL}_{hZ}$ corresponding to the new error sequence Z. The SNR of Z is between the other two SNRs of X and Y error sequences.

Once the $\mathbf{NEL}_{hZ}$ is calculated, the set of steps described before to construct the submodels are applicable in the process towards generating the required error sequence. The new error sequence Z can then be generated directly using $\mathbf{NEL}_{hZ}$ without the need for a reference error sequence. Therefore, the key to generate new error sequences

Table 5.2: The ABWHMM algorithm.

1. Find $\mathbf{NEL}_h$ vectors and error-free burst distribution corresponding to error sequence X.
2. Find $\mathbf{NEL}_h$ vectors and error-free burst distribution corresponding to error sequence Y.
3. Sort the contents of each vector in descending order.
4. Sort each set of vectors related to X and Y in descending order, i.e., according to the first element in each vector $\mathbf{NEL}_h$.
5. Work out the $\mathbf{NEL}_h$ corresponding to the required error sequences Z using (5.9), (5.10), and (5.11).
6. Follow the normal steps of creating the HMM submodels to generate the error bursts.
7. Work out the error-free distribution corresponding to the new error sequence Z by interpolation between those of X and Y.
8. Derive the new error-free bursts and concatenate them to the new error bursts.

with any required SNRs from two reference error sequences using the ABWHMM is to obtain the $\mathbf{NEL}_{hZ}$s of new error sequences.

In order to find $\mathbf{NEL}_{hZ}$, we need to first sort each $\mathbf{NEL}_{hX}$ and $\mathbf{NEL}_{hY}$ in a descending order so that the PNE is the leading element. Secondly, $\mathbf{NEL}_{hX}$ and $\mathbf{NEL}_{hY}$ each should be sorted in the vertical direction, e.g., the PNE for NEL_{1X} is greater than PNE for NEL_{2X} and so on. Then, $\mathbf{NEL}_{hZ}$ can simply be obtained by

$$\mathbf{NEL}_{hZ} = \lfloor \alpha \cdot \mathbf{NEL}_{hX} + \beta \cdot \mathbf{NEL}_{hY} \rfloor. \tag{5.11}$$

Here, $\lfloor \cdot \rfloor$ denotes the floor function and α and β are given by (5.9) and (5.10), respectively. Afterwards, we apply the classification rule and training procedure to generate new error bursts.

To generate error-free bursts related to the new error sequence Z, we interpolate two error-free burst distributions related to error sequences X and Y. Moreover, the number of error-free bursts of Z is obtained by interpolation from the numbers of error-free bursts of X and Y. Subsequently, the error-free burst lengths are worked out and substituted with zeros. Eventually, concatenation of new error bursts and error-free bursts is performed to produce the error sequence Z.

5.4 Adaptive DPBGM (ADPBGM)

In the DPBGM, the $\mathbf{EB}_{rec}$ and $\mathbf{EFB}_{rec}$ are collated as vectors in order to generate error sequences at a later stage. We denoted the minimum value in $\mathbf{EB}_{rec}$ as m_{B1} and the maximum value as m_{B2}. Moreover, the minimum value and the maximum value in $\mathbf{EFB}_{rec}$ were denoted as $m_{\bar{B}1}$ and $m_{\bar{B}2}$, respectively. For the convenience of developing the ADPBGMs, let us remember two quantities:

1) $N_{EB}(m_e)$ 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. Then, the EBD $P_{EB}(m_e)$ can be calculated

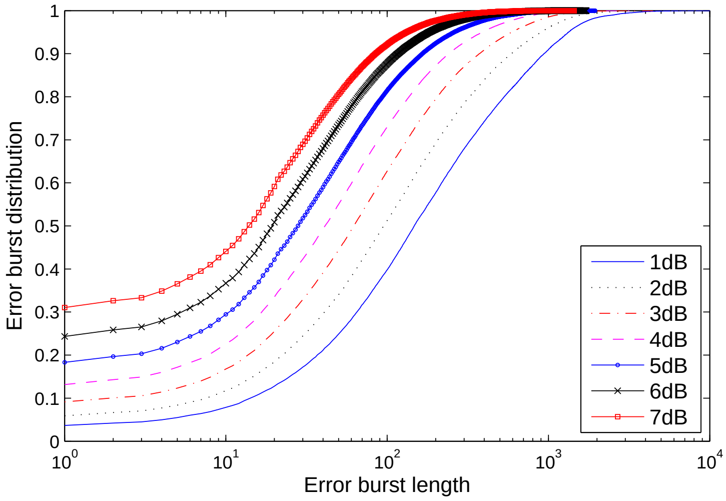

FIGURE 5.2: EBDs of the descriptive model obtained from the LTE system at different SNRs.

by

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

2) $N_{EFB}(m_{\bar{e}})$ 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. Then, the EFBD $P_{EFB}(m_{\bar{e}})$ is given by $P_{EFB}(m_{\bar{e}}) = \frac{1}{\mathcal{N}_{EFB}} \sum_{x=m_{\bar{B}1}}^{m_{\bar{e}}} N_{EFB}(x)$.

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. TABLE 5.3 illustrates the ADPBGM method.

Subsequently, the key to generate a new error sequence Z with any required SNRs from two reference error sequences X and Y, having different SNRs, i.e., SNR$_X$ and SNR$_Y$, is to obtain the parameters $\psi = (\sigma_0, f_{max}, T_A)$, $\tilde{P}_{EB}$, $\tilde{P}_{EFB}$, $\hat{N}_{EB}(m_e)$, and $\hat{N}_{EFB}(m_e)$ for the error sequence Z from similar parameters of the error sequences X and Y.

The first key parameter ψ is calculated by

$$\psi_Z = \alpha \cdot \psi_X + \beta \cdot \psi_Y. \tag{5.12}$$

The values of α and β are given by (5.9) and (5.10), respectively. Then, we have to find $\tilde{P}_{EB_Z}$ of the error burst generator by

$$\tilde{P}_{EB_Z} = \alpha \cdot \tilde{P}_{EB_X} + \beta \cdot \tilde{P}_{EB_Y}. \tag{5.13}$$

In order to construct $\widetilde{\mathbf{EB}}_{rec}$ from $\tilde{P}_{EB_Z}$, we have to know the total number of error bursts $\tilde{\mathcal{N}}_{EB}$ for the error sequence Z. The number $\tilde{\mathcal{N}}_{EB_Z}$ is obtained by interpolating between $\tilde{\mathcal{N}}_{EB_X}$ and $\tilde{\mathcal{N}}_{EB_Y}$, given that X and Y have the same length. Multiplying the obtained number with the extracted $\tilde{P}_{EB_Z}$ with some manipulations related to the CDF gives us the required $\widetilde{\mathbf{EB}}_{rec}$ of Z. Similar procedure can be applied to find $\widetilde{\mathbf{EFB}}_{rec}$. In order to apply the mappers, we need to find $\hat{N}_{EB}(m_e)$ corresponding to Z by finding P_{EB_Z} as follows

$$P_{EB_Z} = \alpha \cdot P_{EB_X} + \beta \cdot P_{EB_Y}. \tag{5.14}$$

The distribution P_{EB} has a monotonically increasing property which simplifies finding new curves of P_{EB} between the other two P_{EB} curves. FIGURE 5.2. demonstrates this property of P_{EB}. Now, we have to work out the total numbers of error bursts

TABLE 5.3: The ADPBGM algorithm.

1. Find ψ, $\tilde{P}_{EB}$, $\tilde{P}_{EFB}$, $\hat{N}_{EB}(m_e)$, and $\hat{N}_{EFB}(m_e)$ corresponding to error sequence X.
2. Find ψ, $\tilde{P}_{EB}$, $\tilde{P}_{EFB}$, $\hat{N}_{EB}(m_e)$, and $\hat{N}_{EFB}(m_e)$ corresponding to error sequence Y.
3. Work out ψ, $\tilde{N}_{EB}(m_e)$, $\tilde{N}_{EFB}(m_e)$, the modified $\tilde{N}_{EB}(m_e)$, and the modified $\tilde{N}_{EFB}(m_e)$ corresponding to error sequence Z from (5.9), (5.10), (5.12), (2.7), (2.8), (5.13), and (5.14).
4. Find out the modified $\mathbf{EB}_{rec}$ and $\mathbf{EFB}_{rec}$ corresponding to the required error sequences Z.
5. Create the error bursts and error-free bursts corresponding to the error sequences Z.
6. Follow the normal procedure of the DPBGM to generate the error sequence Z.

$\mathcal{N}_{EB_Z}$ for the error sequence Z by interpolating between $\mathcal{N}_{EB_X}$ and $\mathcal{N}_{EB_Y}$ given that X and Y have the same length, and then we find $\hat{N}_{EB}(m_e)$. Similar methodology can be applied to find $\hat{N}_{EFB}(m_e)$. Consequently, mappers are used according to (2.7) and (2.8) to work out the modified $\widetilde{\mathbf{EB}}_{rec}$ and $\widetilde{\mathbf{EFB}}_{rec}$.

Afterwards, we can create error bursts and error-free bursts according to the lengths in the modified $\widetilde{\mathbf{EB}}_{rec}$ and $\widetilde{\mathbf{EFB}}_{rec}$. Generating error-free bursts is simple because the lengths of the modified $\widetilde{\mathbf{EFB}}_{rec}$ can easily be 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 error sequences X and Y. The retrieved error bursts have the same length m_e as in the newly obtained $\widetilde{\mathbf{EB}}_{rec}$ of Z. Finally, the error bursts and error-free bursts are combined together to construct the new required error sequence Z.

5.5 Adaptive DPB-HMM (ADPB-HMM)

In order to design the ADPB-HMM, we follow the same procedure aforementioned in Section 5.4. However, the generation stage of the DPB-HMM is different and depends on the DEPHMM. Therefore, we have to focus on the transition matrix as well as the distribution of the gaps of the second layer of the HMM. This is because the second layer of DEPHMM is in charge of constructing the error bursts. Constructing adaptive error-free bursts has been discussed in Section 5.4. Thus, the main work here is to construct the adaptive error bursts. The concatenation between the adaptive error-free bursts and adaptive error-bursts is subsequently conducted in the first layer to generate new adaptive error sequences. Table 5.4 shows the detailed steps of ADPB-HMM.

The first step is to calculate the two transition matrices d_X^u and d_Y^u of the error cluster substates in the second layer in DEPHMM for two different SNRs corresponding to two reference error sequences X and Y, respectively. Our aim is to find the substates transition matrix d_Z^u for the adaptive error sequence Z.

We perceived that each row, corresponding to each error burst state, in the substates transition matrix, i.e., the substate transition vector in each error burst state, is monotonically decreasing with the number of substates. Therefore,

$$d_Z^u = \alpha \cdot d_X^u + \beta \cdot d_Y^u \tag{5.15}$$

The values of α and β are given by (5.9) and (5.10), respectively.

It is also very important to know the number of error cluster substates in each error burst state. It is already known that the number of error cluster substates corresponds to the maximum error cluster length in each error burst state. The new number of the error cluster substates for Z is worked out by

$$M_Z^u = \lfloor \alpha \cdot M_X^u + \beta \cdot M_Y^u \rfloor \tag{5.16}$$

The error cluster substates are now constructed. Nonetheless, to complete the second layer design, we need to find the gaps substate, for each error burst state, corresponding to error sequence Z. We have found by investigations that the gaps distribution in

TABLE 5.4: The ADPB-HMM algorithm.

```
1. Find out the modified $\mathbf{EB}_{rec}$ and $\mathbf{EFB}_{rec}$ corresponding to
    the required error sequences $Z$ as in TABLE 5.3.
2. Find $d_X^u$, $M_X^u$, and $\mu_X^u$ corresponding to error sequence $X$.
3. Find $d_Y^u$, $M_Y^u$, and $\mu_Y^u$ corresponding to error sequence $Y$.
4. Work out $d_Z^u$, $M_Z^u$, and $\mu_Z^u$ corresponding to error sequence $Z$
    from (5.9), (5.10), (5.15), (5.16),and (5.17).
5. Create the new error bursts.
6. Work out $P_{EFB}$ corresponding to error sequence $Z$.
7. Work out $N_{EFB}$ corresponding to error sequence $Z$.
8. Find out the $\mathbf{EFB}_{rec}$ corresponding to the required error sequences $Z$.
9. Create the new error-free bursts.
10. Concatenate together the new error bursts and error-free bursts.
```

the second layer follows the exponential distribution. However, to define any exponential distribution, its mean value is required. The mean values of the gaps distributions related to error sequence Z is given by

$$\mu_Z^u = \alpha \cdot \mu_X^u + \beta \cdot \mu_Y^u \tag{5.17}$$

The gap substate as well as the error clusters substates are now ready for each error burst state of the new error sequence Z. Consequently, the error bursts can now be constructed in the second layer and can subsequently be concatenated with the error-free bursts in the first layer to form the new error sequence Z.

5.6 Simulation Results and Discussions

To validate our proposed adaptive generative models, we first need to obtain reference error sequences, which are essential to initialise various parameters for the generative models. We use an uncoded LTE system to obtain the required reference error sequences.

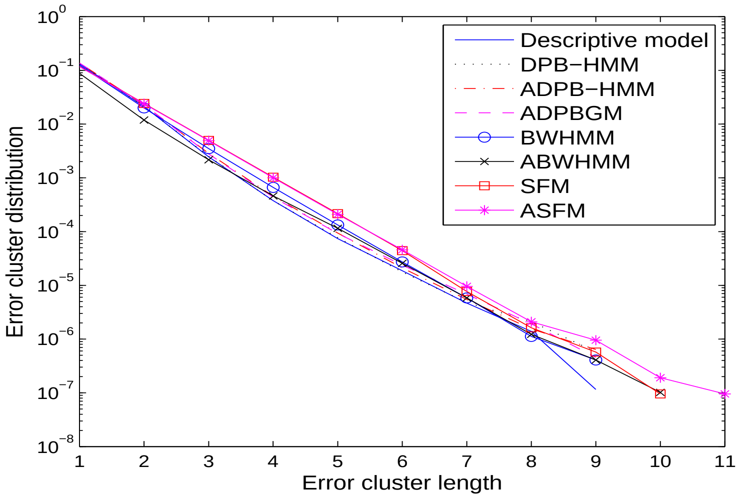

FIGURE 5.3: ECDs of the descriptive model and different adaptive generative models.

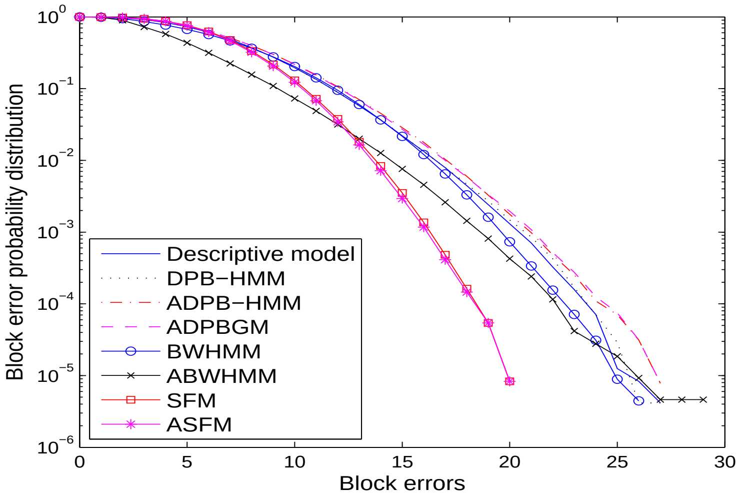

FIGURE 5.4: BEPDs of the descriptive model and different adaptive generative models.

The open source Vienna LTE simulator v1.6r917 [89–91] that is used here, includes Adaptive Modulation and Coding (AMC), MIMO transmissions, downlink transmission scheme based on Orthogonal Frequency Division Multiple Access (OFDMA). The LTE simulator can also be utilised at both link level and system level. We used the link level simulator which consists of one transmitting eNodeB, one receiver User

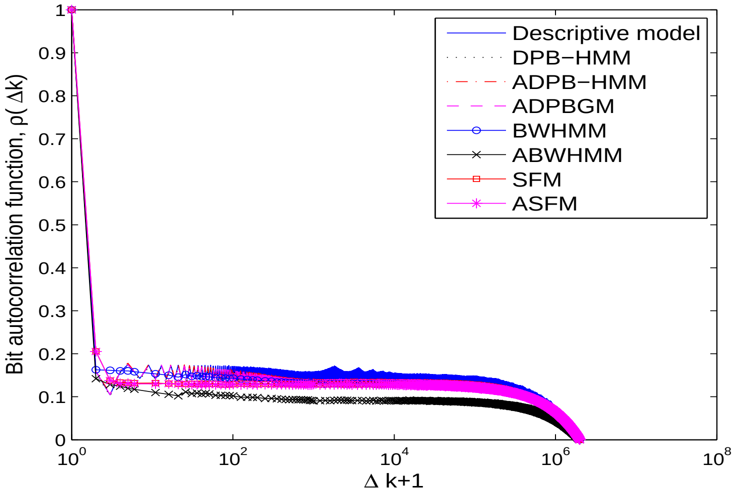

FIGURE 5.5: BCFs of the descriptive model and different adaptive generative models.

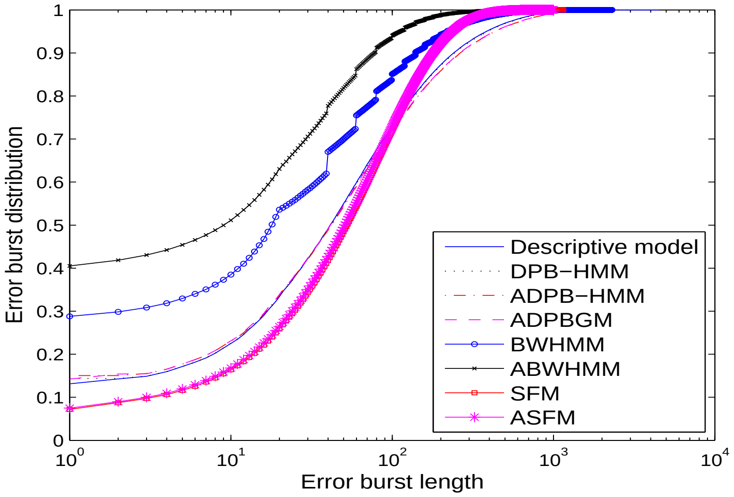

FIGURE 5.6: EBDs of the descriptive model and different adaptive generative models.

Equipment (UE), a downlink channel model over which only the Downlink Shared Channel (DL-SCH) is utilised, signalling information, and an error-free uplink feedback channel with zero delay. We used the following channels: RA275, TU3, TU50, PedB5 (pedestrian B)5 , and PedB10. The received data has a length of 12×10^6 and the transmission rate is $F_s = 3450$ kb/s. Reference error sequences were produced at SNRs of 1 dB, . . . , 7 dB.

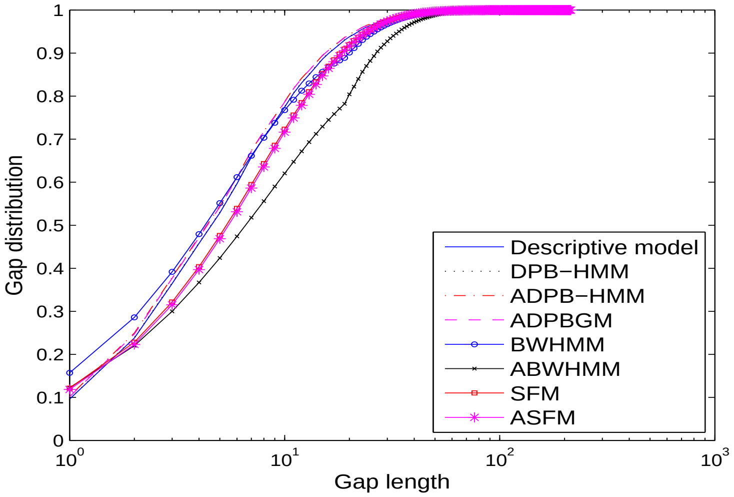

FIGURE 5.7: GDs of the descriptive model and different adaptive generative models.

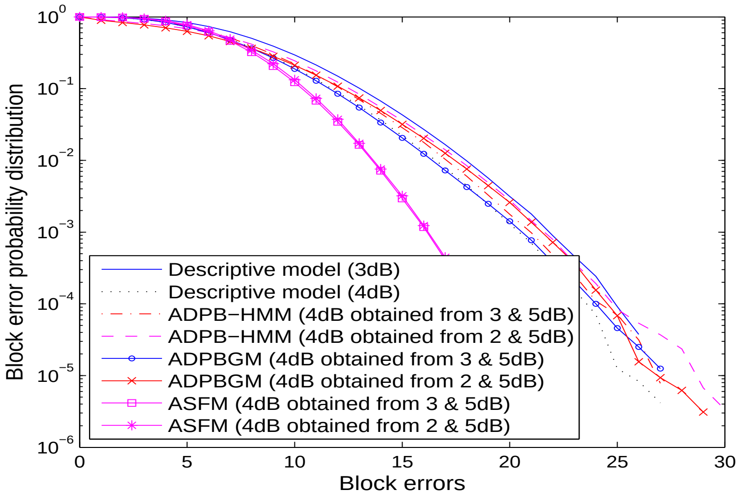

Figure 5.8: BEPDs of the descriptive model and two adaptive generative models ($n = 50$).

By comparing the transmitted error sequence with the received one, we work out the bit error sequences. We use the four discussed generative models, namely, the ASFM, ABWHMM, ADPBGM, and ADPB-HMM in order to generate new error sequences of length 15×10^6 bits based on the obtained error sequences from the LTE system. Here, we show only the results of the TU50 channel having SNRs of 2 dB, 3 dB, 4 dB, and 5 dB.

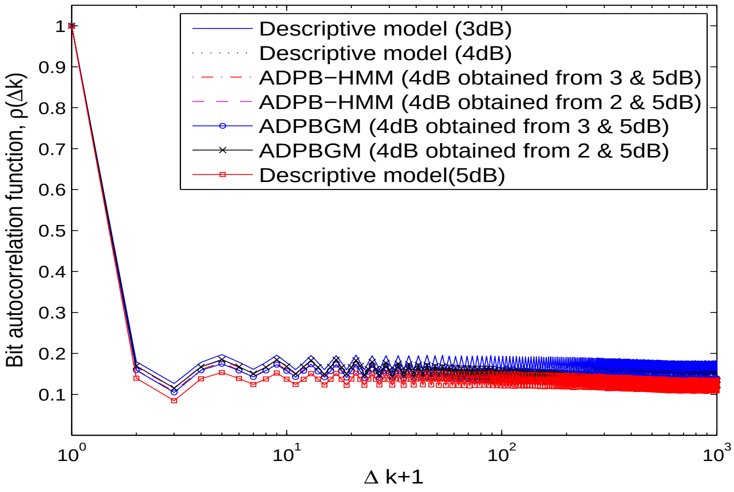

Figure 5.9: BCFs of the descriptive model and ADPBGMs.

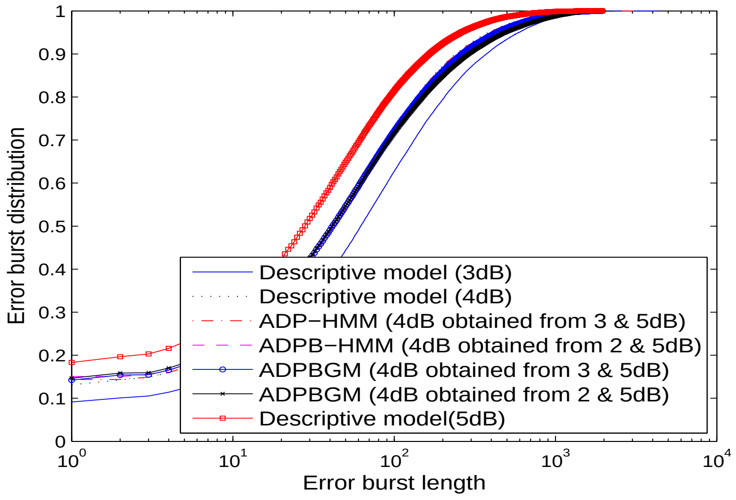

FIGURE 5.10: EBDs of the descriptive model and ADPBGMs.

The performance criteria when comparing between different generative models are always the burst error statistics. A generative model is better if its burst error statistics better match those of the descriptive model, especially the most important statistics such as the BEPD and BECF.

For the sake of comparison, the fitting of $P(0^{m_0}|1)$ in ASFM is achieved by using five exponentials and therefore, $N = 6$ holds. After we fit (5.3) with $P(0^{m_0}|1)$ of

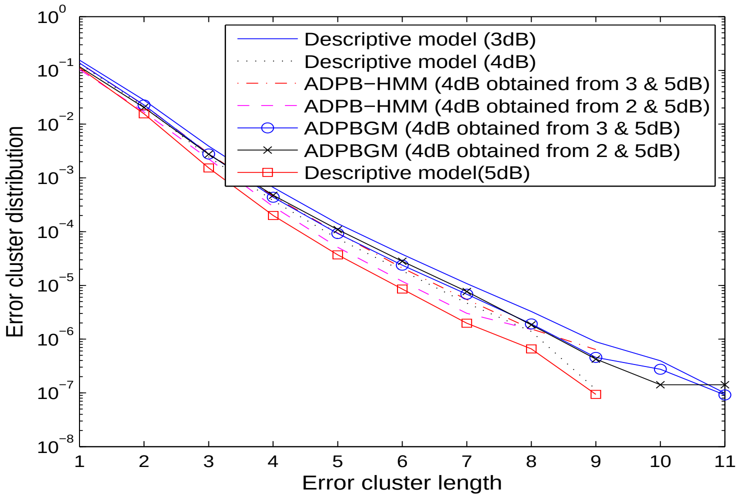

FIGURE 5.11: ECDs of the descriptive model and ADPBGMs.

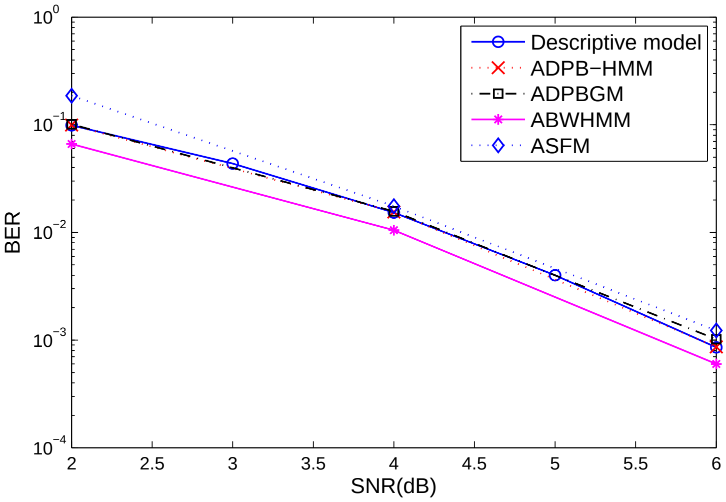

FIGURE 5.12: The coded BER of the descriptive model and three adaptive generative models.

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 (5.9), (5.10), and (5.8) to calculate the adaptive $P(0^{m_0}|1)$ having SNR of 4 dB. Once we know the adaptive $P(0^{m_0}|1)$, (5.3) and (5.1) can be applied to generate the new error sequences.

For the ABWHMM, we 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 and obtain the $\mathbf{NEL}_h$ vectors. We apply (5.11) afterwards to obtain the $\mathbf{NEL}_{hZ}$ vector. A Baum-Welch training process will then be applied to $\mathbf{NEL}_{hZ}$ after classifying it into a satisfactory number of states. The number of classes N is 7 in our example, and 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.

In order to proceed with the ADPBGM and ADPB-HMM, we need to find the vector $\boldsymbol{\Psi}$. The value of q_s was 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 $\mathcal{R}_B = 8.43, 5.24, 2.09$, respectively. Consequently, $\boldsymbol{\Psi}$ = (9, 10, 0.09, 0.0425, 34.9 kHz, 8.33 μs) for SNR = 2 dB, $\boldsymbol{\Psi}$ = (9, 10, 0.09, 0.0470, 36.8 kHz, 5.43 μs) for SNR =

3 dB, and $\mathbf{\Psi} = (9, 10, 0.09, 0.0599, 40.9\text{ kHz}, 4.97\ \mu\text{s})$ for SNR = 5 dB. For the new required error sequence Z, the vector $\mathbf{\Psi}_Z$ can be obtained by (5.12). Then, we find $\tilde{N}_{EB}(m_e)$ and the modified $\tilde{N}_{EB}(m_e)$ for the error sequence Z with the aid of (5.13) and (5.14). Similarly, $\tilde{N}_{EFB}(m_e)$ and the modified $\tilde{N}_{EFB}(m_e)$ can be found. Eventually, the error bursts and error-free bursts are combined together to construct the entailed error sequences. The bit structure in the new error bursts is retrieved from the other two surrounding error sequences based on the error burst lengths. For the ADPB-HMM the number of states is 20. We calculate (5.15), (5.16), and (5.17) in order to create the error bursts and then we concatenate them with the error-free bursts in order to generate the new error sequences.

In our simulations, we produce an error sequence Z of 4 dB SNR from reference error sequences X of 3 dB and Y of 5 dB SNRs (First scenario, FIGURES 5.3–5.7). We also produce an error sequence Z with SNR=4 dB from error sequences X with SNR=2 dB and Y with SNR=5 dB (second scenario, Figs. FIGURES 5.8–5.11). This means that $\alpha = \beta = 0.5$ for the first scenario and $\alpha = 1/3$ and $\beta = 2/3$ for the second scenario. The second scenario investigates the impact of further distancing the SNRs of error sequences X and Y from the SNR of error sequences Z. We compare the burst error statistics of error sequences Z, obtained from both scenarios, with those statistics obtained from the reference error sequence of the LTE simulator having SNR=4 dB.

In terms of parametrisation, the value of η can be found from FIGURE 5.1 when the curve is tending to turn and it was chosen to be 20.

FIGURES 5.3–5.11 illustrate various burst error statistics of the descriptive model and adaptive generative models such as the ECDs (FIGURE 5.3 and FIGURE 5.11), BEPDs (FIGURE 5.4 and FIGURE 5.8), BCFs (FIGURE 5.5 and FIGURE 5.9), EBDs (FIGURE 5.6 and FIGURE 5.10), and GDs (FIGURE 5.7). FIGURES 5.3–5.7 omit the comparison between the DPBGM and ADPBGM since the DPBGM gives approximate results to the descriptive one[70].

In general, the ADPBGM shows the best fit to the descriptive model. This is clear for all the burst error statistics except a small mismatch at the end of the curve of the ECD (FIGURE 5.3) and BEPD (FIGURE 5.4). The second best generative model is the ASFM, although its burst error statistics considerably mismatch those of the ADPBGM. However, the ASFM and SFM fit each other. The ABWHMM results are slightly worse than those obtained using the normal BWHMM procedure.

As it was noticed, the ADPBGM burst error statistics match those of the DPBGM and those of the ASFM match those of the SFM. In addition, this applies to DPB-HMM and ADPB-HMM because their main characteristics, i.e., EBD, EFRD, and d^u, respectively, have a certain known trend as a monotonically increasing or decreasing. However, the ABWHMM main characteristic is a vector which is difficult to be utilised for deriving a new accurate vectors using the interpolation methods.

We also examine the ADPBGM and ADPB-HMM by producing error sequences of 4 dB from other error sequences of 2 and 5 dB as shown in Figures 5.8–5.11 (second scenario). We omit the ABWHMM investigations from the second scenario because their burst error statistics do not match those of the BWHMM for the first scenario, knowing that the second scenario is distancing the reference error sequences more from the new error sequences. It is found that, distancing the SNRs that are required to produce the new error sequence, deteriorates the performance. Figure 5.8 also illustrates the production of 4 dB 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 parametrise the ASFM can be obtained by any pair of other EFRDs of different SNRs. However, the general performance of the ASFM is not satisfactory. Figures 5.8–5.11, show that the ADPBGM and ADPB-HMM are slightly affected by changing α and β. Figure 5.12 shows the coded BER curves after feeding the generated error sequences obtained from neighbouring error sequences X and Y with SNRs distancing one unit from the SNR of error sequences Z. It is apparent that the ADPBGM outperforms the other models.

5.7 Summary

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 this chapter, 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 computational 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. Two reference error sequences with 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 followed by the ADPB-HMM can approximately fit the descriptive model. Other adaptive generative models like the ASFM and AB-WHMM give poor burst error statistics compared to the descriptive model. However, the ASFM is superior to the ABWHMM regarding most of the burst error statistics. In other words, the ASFM performance is generally closer to that of the descriptive model than the one of the ABWHMM. It has also been 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 neighbouring error sequences. In contrary, the ADPB-HMM, ABWHMM, and ASFM can create the error bursts and error-free bursts automatically once the required parameters are calculated.

Chapter 6

Applications of Error Models to HARQ

6.1 Introduction

Wireless communication systems employ HARQ in order to effectively detect and correct errors occurred in wireless channels and hence enhance the throughput performance of the system. HARQ consists of error detection, Forward Error Correction (FEC), and the well-known ARQ, e.g., an N-channel Stop-And-Wait (SAW) protocol [92]. There are three types of HARQ, namely packet combining or Chase Combining (CC), full Incremental Redundancy (IR), and partial IR [93]. By using the Cyclic Redundancy Check (CRC), erroneous packets can be detected and a request for a retransmission is sent to the transmitter. The retransmission can be a duplicate packet or just some redundancy bits that are combined with the erroneous packet so that it can easily be corrected by the FEC. For instance, the LTE system utilises the full IR HARQ with 1/3 turbo encoder [94]. Full IR HARQ decreases the coding gain in each retransmission by retransmitting only redundancy bits, which will be combined with the stored erroneous packet in the receiver buffer. The retransmission continues when needed until the packet is successfully decoded at the receiver or the maximum number of retransmissions is reached. In the latter case, the packet is discarded and is recorded as a packet error.

In this chapter, we obtain packet error sequences utilising the Vienna LTE simulator [89, 90]. We first statistically analyse the obtained error sequences without considering HARQ. We then produce more packet error sequences applying the adaptive methods. These newly obtained error sequences replace the entire physical layer and feed the HARQ protocol. At last, Packet Error Rate (PER) performance is obtained. In the second part of this chapter, we investigate more statistical properties of the HARQ error sequences for a different type of channel. Moreover, we propose a prediction generative model that is capable of creating packet error sequences with similar burst error statistics to those obtained from the LTE Vienna simulator having HARQ from error sequences that do not take into account the effect of HARQ. In other words, the newly developed prediction generative model can predict the performance of HARQ in terms of high-order statistics called burst error statistics. In the literature [95–102], researchers have predicted the performance of HARQ in terms of PER only, which is a first-order statistic. However, we here predict higher order statistics of the HARQ performance.

6.2 Applications of Adaptive Generative Models to HARQ

As previously mentioned, for this section, we adaptively generate packet error sequences in order to replace the whole physical layer with them. These adaptive packet error sequences are used to test the HARQ performance in terms of the PER.

6.2.1 Simulation Results and Discussions

In our simulations, we use again the LTE simulator. The detailed data processing in the transmitter and receiver is as follows. In the transmitter, the user data are

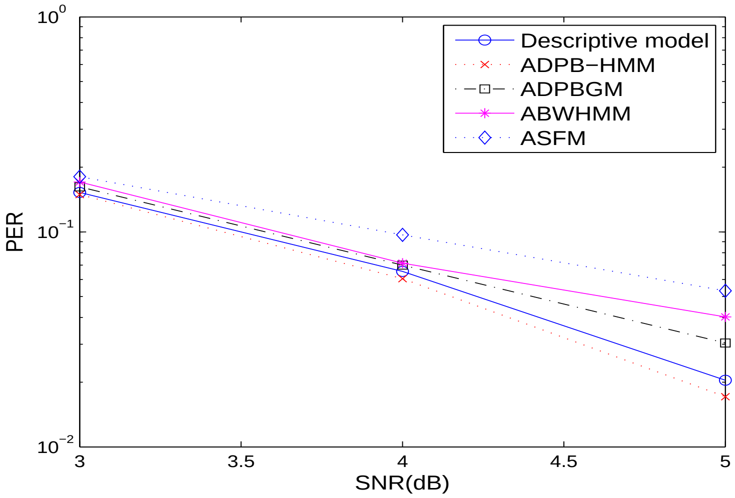

FIGURE 6.1: The PER of the descriptive model and three adaptive generative models after the HARQ.

prepared as Transport Blocks (TBs). A CRC is derived and appended to each TB. Then, each TB is encoded using a turbo encoder, interleaved, and rate-matched with a target rate depending on the received CQI. After these processes, the blocks are modulated and then mapped to the transmission antennas. The LTE simulator can utilise several channel models based on ITU and 3GPP standards.

In the receiver, each UE receives the signals transmitted by the eNodeB and performs signal processing in order to extract the useful transmitted data. Signal demapping, demodulation, and decoding are executed. The CRC of the received packets is calculated in order to check whether the packets are received in errors or correctly. If the received packet is correct, then “0” is recorded in the error sequence. Otherwise, “1” is noticed in the error sequence.

We use here a TU100 channel with SNRs of 2 dB, 3 dB, 4 dB, 5 dB, and 6 dB. The length of the obtained error sequences is 0.5 million packets. These error sequences are exploited to generate more of them by means of developed adaptive methods. These newly generated packet error sequences are then fed into the HARQ protocol to test its performance in terms of PER.

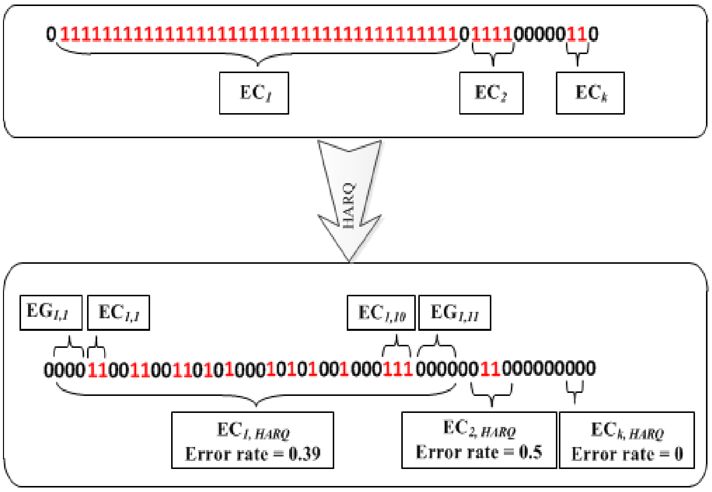

FIGURE 6.2: An error sequence extract to show the effect of adding the HARQ.

FIGURE 6.1 shows the PER of the descriptive model and ADPB-HMM, ADPBGM, ABWHMM, and ASFM at 3 dB, 4 dB, and 5 dB. The adaptive packet error sequences are obtained utilising two neighbouring error sequences distancing from the required SNR by one unit at both sides. FIGURE 6.1 illustrates that the ADPB-HMM and the ADPBGM have the closest performance to the descriptive model. The ABWHMM comes next followed by the ASFM.

6.3 Predicting the Burst Error Statistics of HARQ

In order to develop the prediction generative model [103], we first have to understand the effect of HARQ on error sequences, which are obtained without HARQ. Therefore, we obtain error sequences with disabled and enabled HARQ from the LTE simulator and compare between them. FIGURE 6.2 shows the effect of HARQ on an extract of an error sequence. The HARQ breaks the error clusters reducing the error correlation and producing new error rates inside the error cluster blocks of lengths $\mathbf{EC}_i$ $(\mathbf{EC}_1, ..., \mathbf{EC}_k)$, k is the number of error clusters of the original error sequence that does not include HARQ. This property of having many shorter error clusters after including the HARQ, guides us to develop a prediction generative model that

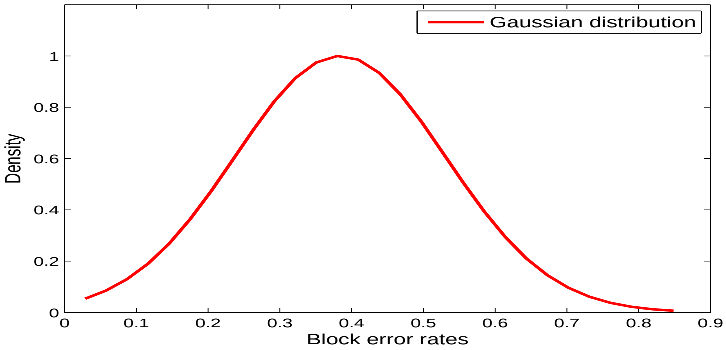

FIGURE 6.3: The density of error rates inside $\mathbf{EC}_i$ after using the HARQ.

can predict the performance of HARQ. In other words, the prediction generative model generates error sequences that cover the physical layer with HARQ from error sequences obtained without considering the HARQ. In this way, we can catch the statistical behaviour of the HARQ.

To characterise the effect of adding HARQ to the physical layer, the following steps are considered:

1. we extract the k error clusters from the original error sequence which does not include HARQ.
2. we work out the error cluster lengths recorder $\mathbf{EC}_i$ and gap lengths recorder $\mathbf{G}_j$ $(\mathbf{G}_1, ..., \mathbf{G}_{k+1})$.
3. we now consider the effect of HARQ. That means some errors are corrected in each $\mathbf{EC}_i$. Therefore, each original error cluster converts to new smaller error clusters, and new gaps are formed as previously mentioned. In other words, the error rate inside each $\mathbf{EC}_i$ reduces from 1 to a value less than 1. We call the new error cluster lengths $\mathbf{EC}_{i,h}$ $(h = 1, ..., l_i)$, and new gap lengths $\mathbf{EG}_{i,u}$ $(u = 1, ..., l_i + 1)$. Here, l_i indicates the number of new error clusters in each $\mathbf{EC}_i$.

The aforementioned error rate distribution follows a Gaussian distribution $\mathcal{N}$ (μ, σ^2). We have tested that for different channels, such as the TU, Pedestrian A (PedA), and Pedestrian B (PedB) channels. FIGURE 6.3 demonstrates the new error rates distribution for PedB. The mean value μ of the Gaussian distribution is the PER at the specific SNR, such that

$$\mathcal{N}(\mu, \sigma^2) = \mathcal{N}(\text{PER}, \sigma^2). \tag{6.1}$$

The PER can be calculated by using one of the PER prediction methods in [95–98]. These methods do not produce error sequences but predict the PER curves by other mathematical means. For example, we use the Exponential Effective SIR Mapping (EESM) algorithm in [95] to work out the required PER. The basic idea of EESM is to calculate the instantaneous effective SNR δ_{eff} of the AWGN channel that yields the right PER at the given SNR value δ in the LTE wireless channel, such that

$$\text{PER}(\delta) = \text{PER}_{\text{AWGN}}(\delta_{eff}) \tag{6.2}$$

where

$$\delta_{eff} = -\gamma \ln\left(\frac{1}{\mathcal{N}} \sum_{k=1}^{\mathcal{N}} e^{\left(\frac{-\delta_k}{\gamma}\right)}\right) \tag{6.3}$$

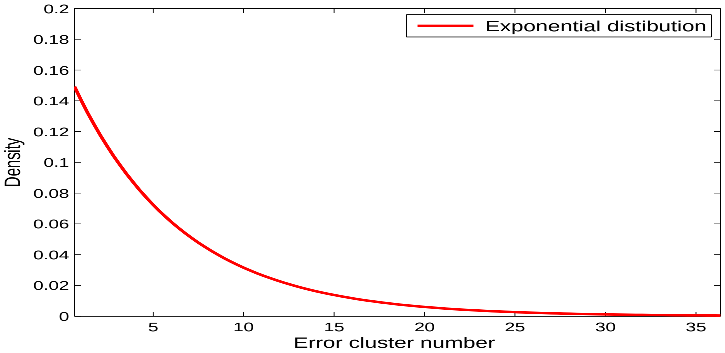

FIGURE 6.4: The density of error rates inside $\mathbf{EC}_i$ after using the HARQ.

and $\mathcal{N}$ is the number of subcarriers, γ is an optimisation parameter depending on the code rate, modulation, and block size. Note that this method requires the prior knowledge of PER curve of the system with AWGN channel.

Figure 6.4 also shows that the number of new error clusters l_i within $\mathbf{EC}_i$ after the retransmission follows the exponential distribution. The mean value of the exponential distribution is:

$$\epsilon = \frac{\sum_{i=1}^{k} \mathrm{EC}_i}{k \times \delta}. \tag{6.4}$$

Here, $\delta \leq 10$dB in our examples in order to guarantee sufficient number of error clusters (k) with sufficient error cluster lengths ($\mathbf{EC}_i$).

As we have recognised the distributions of both the new error rates and the error cluster lengths after the retransmission, we are able to produce any quantity of these error rates and error cluster lengths to construct generated error clusters combined with gaps that together occupy lengths equivalent to $\mathbf{EC}_i$ of the system with HARQ. We rename these lengths as $\mathbf{EC}_{i,HARQ}$. In other words, we produce $\mathbf{EC}_{i,HARQ}$ to replace $\mathbf{EC}_i$ of the original error sequence as of the effect of including the HARQ (see Figure 6.2).

The procedure of constructing the $\mathbf{EC}_{i,HARQ}$ is as follows.

1. we randomly choose an error cluster length $\mathbf{EC}_i$.
2. we choose the number of new error clusters l_i (obtained from the exponential distribution) that can fit within the chosen $\mathbf{EC}_i$ according to a specified error rate (obtained from the Gaussian distribution).
3. we fill in random gaps between these error clusters to complete the required length $\mathbf{EC}_i$, giving that the first and last parts within $\mathbf{EC}_i$ are gaps. Hence, the structure of the $\mathbf{EC}_{i,HARQ}$ is completed.

To finalise the process of generating a new error sequence that takes into account the effect of HARQ, we replace each $\mathbf{EC}_i$, which contains ones only, with one of

$\mathbf{EC}_{i,HARQ}$, which now contains zeros and ones. Consequently, the gap and error cluster distributions are now completely different from those obtained from the original error sequence.

6.3.1 Simulation Results and Discussions

The same LTE simulator was used here to obtain reference error sequences with the existence and absence of HARQ. When HARQ is included, if the received packet is correct, an acknowledgment signal is sent back to the transmitter. Otherwise, the transmitter will send further information, with the help of the HARQ scheme, in order to assist the receiver in correcting the errors. The received packet is dropped once the retransmission fails to correct the errors. The receiver is also capable of estimating the channel from the received data using the reference signals. From the channel estimation, the quality of the channel is evaluated and feedback information is sent in order to help the transmitter cope with the channel impairments by adjusting the transmission parameters.

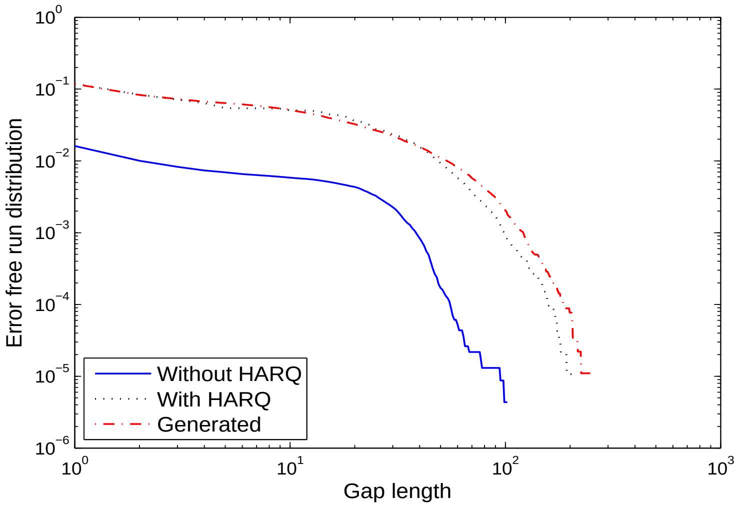

FIGURE 6.5: EFRDs of LTE error sequences with and without HARQ and the HARQ prediction generative model..

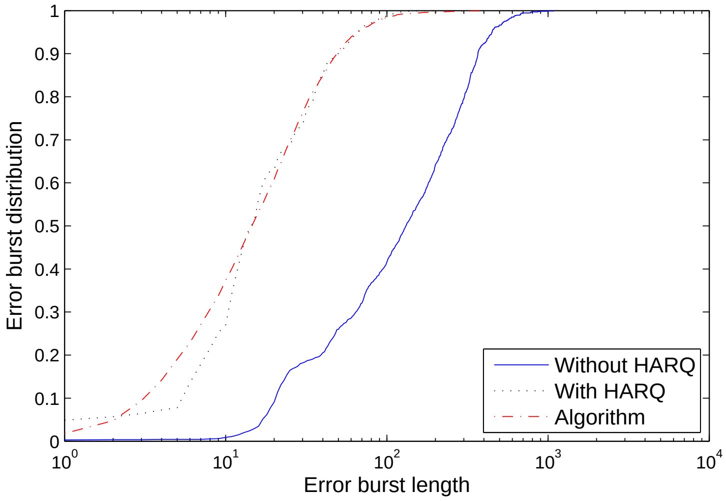

FIGURE 6.6: EBDs of LTE error sequences with and without HARQ and the HARQ prediction generative model.

In the LTE simulator, we use pedB wireless channel with an average SNR of 10 dB. The user speed is set at 5km/h, the bandwidth is 1.4MHz, and the CQI is set to be 7, 8, and 9. This means that the modulation scheme used is 16QAM with coding rates 0.37, 0.48, and 0.60. Other modulation schemes are also tested. The MIMO configuration is 2×1. We show here only the results having CQI of 8. The number of

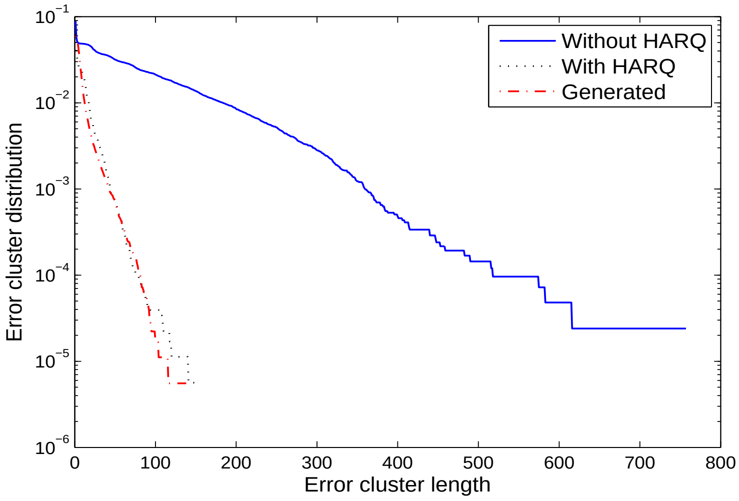

FIGURE 6.7: ECDs of LTE error sequences with and without HARQ and the HARQ prediction generative model.

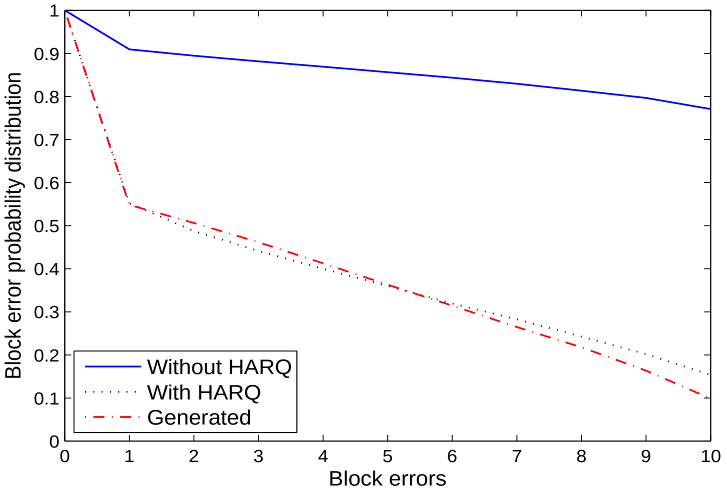

FIGURE 6.8: BEPDs of LTE error sequences with and without HARQ and the HARQ prediction generative model ($n = 10$).

transmitted packets is 500000. For our used parameters, the method of EESM gives us that the PER is 0.38 at 10 dB. The variance in (6.1) is chosen to be very small, such as 0.02, this value has not been exceeded in our examples. The mean value of the exponential distribution is calculated as 6.

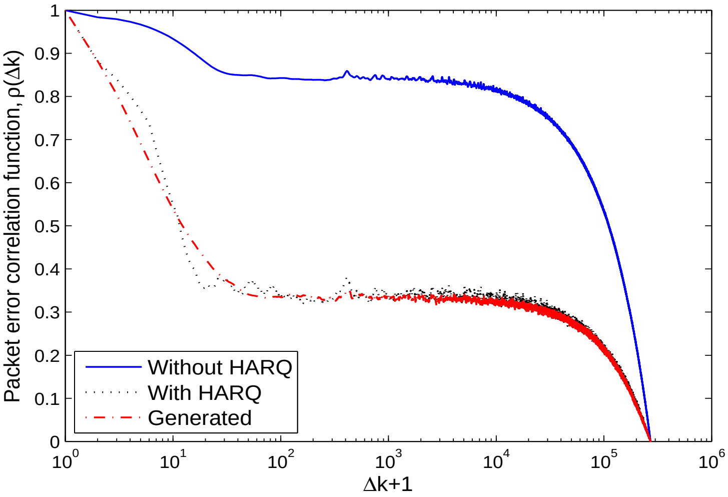

FIGURE 6.9: PECFs of LTE error sequences with and without HARQ and the HARQ prediction generative model.

Figures 6.5– 6.9 show the burst error statistics with and without HARQ. It is apparent that the error occurrence is severe without HARQ. For example, Figure 6.5 is the probability that an error packet is followed by a specific minimum error-free packets. It is apparent that the gap lengths become longer when the HARQ is added and hence the probability becomes higher. Figure 6.6 shows the error burst distribution and illustrates the extreme change in the error burst lengths when HARQ is introduced. Figure 6.7 demonstrates the error cluster distribution where the length of error clusters is severely affected due to the effect of HARQ. Moreover, many error clusters are cancelled. For Figure 6.8 the number of errors counted in blocks of 10 is very large without HARQ. A dramatic decrease in the number of errors counted in blocks occurs when using HARQ. Figure 6.9 is the most impressive and shows that the error correlation decreases after HARQ.

Figures 6.5– 6.9 also illustrate the predicted burst error statistics of the newly generated error sequence when HARQ is activated. The burst error statistics of the generated error sequence have a satisfactory fit with those burst error statistics of an error sequence obtained directly from the LTE simulator having HARQ included (descriptive model).

It is worth mentioning that receiving 500000 packets in the LTE simulator requires approximately 64 hours using a processor with speed of 2.27 GHz. However, our generated error sequence with the same length as the one obtained directly from the LTE system takes 1.38 s.

6.4 Summary

We have checked the performance of the HARQ using adaptive error sequences. Furthermore, we have proposed a prediction generative model that is capable of predicting the statistical behaviour of HARQ systems in terms of a set of packet-level burst error statistics utilising the predicted PER. The adaptive packet error sequences dramatically reduce the evaluation time of higher layer protocols. We have used the Vienna

LTE open source simulator to obtain packet error sequences. The proposed prediction generative model can generate packet error sequences with burst error statistics very similar to those of error sequences obtained directly from the LTE system with HARQ. Importantly, the proposed prediction generative model is not only accurate but also efficient as it considerably reduces the time consumed for generating HARQ error sequences. Such error sequences are consumables for higher layer checks and performance evaluations.

Chapter 7

Conclusions and Future Work

7.1 Summary of Results

Error models are crucial for the design and performance investigations for many error control strategies as well as higher layer protocols in the data link layer up to the transport layer. Error models can be either descriptive or generative. Descriptive models are derived from real or computer implemented systems for the purpose of parametrisation and comparison. However, generative models play the major role of error models since they construct a handy alternative of descriptive models and hence save the time of designing and/or testing the digital component, scheme, or protocol.

Generative models do exist in the literature, however, they may not be applicable for modern communications systems. Moreover, most of the existing generative models are either not very accurate or extremely complicated for the new wireless communication systems. It is always desirable to have accurate models with simple parametrisation. Compromising between the accuracy and complexity is a key issue in modelling.

In this thesis, we have attempted to develop desired generative models which can cope with new profiles of errors related to widely-used wireless communication systems; and therefore have proposed the DEPHMM and 3LHMM. The former has been tested for hard bit, soft bit, and packet error sequences, and has proven its accuracy and efficiency. The 3LHMM on the other hand improves the generation of required error sequences despite the fact that it has additional complexity. This extra complexity in the setting-up simplifies the data training at the end.

These developed generative models have been compared with the descriptive model as well as the SFM, BWHMM, and DPBGM. In most cases, the SFM and BWHMM provided a non-satisfactory performance compared with the descriptive model. However, the DEPHMM and DPBGM were competing with each other. Although the DPBGM took the lead, it had a structural flaw when generating error bursts which reduced its desirability.

Therefore, a modified version of the DPBGM has been suggested with an aid of the DEPHMM to construct the DPB-HMM. This modification degrades the final performance a bit compared with the DPBGM performance, but increases its desirability. Furthermore, the DPB-HMM performance is still better than the SFM and BWHMM; and the DPBGM remains superior to the SFM and BWHMM at packet-level error profiles.

The DPBGM and DPB-HMM have the best performance because they adopt matching the error burst distribution and error-free burst distribution with those of the descriptive model rather than the error cluster distribution and gap distribution. This is because the error bursts and error-free bursts are the longest components in the error sequence. Consequently, matching their lengths with the reference lengths would implicitly match the lengths of smaller components such as the error cluster and gaps. This will keep the number of errors and their distribution restricted within the error burst length. Subsequently, the other burst error statistics perfectly match the reference statistics. On the contrary, the Markov models generate bit by bit error sequences, and therefore they could not guarantee the correlation between error events. However, the BWHMM demonstrated an improvement over traditional Markov models at the bit level. This is because BWHMM divides the error sequence into blocks and generates error bursts on block basis rather than bit basis. However, there are limitations on the block lengths. The DPB-HMM adopts longer blocks with a different strategy, and hence its results were found to be satisfactory.

Adaptive generative models have been introduced in order to reduce the burden on the descriptive models. The DPBGM still kept the lead position followed by the DEP-HMM. However, the BWHMM performance fell back to the last position preceded by the SFM. In order to further evaluate the performance of the adaptive generative models, generated bit error sequences replaced the digital wireless channel and BER curves were obtained for the physical layer. Moreover, the whole physical layer was replaced by packet error sequences which were produced adaptively in order to further check our adaptive generative models at packet level. The same conclusion was found, that the DPBGM led with a satisfactory performance followed by the DEP-HMM.

Finally, we have also proposed a prediction generative model capable of predicting the statistical behaviour of HARQ systems. The proposed prediction generative model can generate packet error sequences with burst error statistics very similar to those of error sequences obtained directly from the LTE system with HARQ. In fact, the proposed prediction generative model is not only accurate but also efficient as it considerably reduces the time consumed for generating HARQ error sequences. Such error sequences are also consumables for higher layer checks and performance evaluations.

7.2 Future Work

As the research in this area is interesting and has many industrial benefits, it is recommended to further persuade research in this field for further developments and improvements.

One of the further research topics is to conduct more research on adaptive generative models. In our adaptive generative models we have only changed the CIRs and SNRs, but further investigations could be conducted to other parameters of the digital wireless channels such as the modem and coding parameters.

Moreover, the performances of our generative models have been tested against descriptive models of the EGPRS and LTE. However, it is also possible to check the performances of the generative models against other descriptive models, such as the ultra wide band (UWB) or those beyond the 4G systems as well as the wireless sensor networks. Different types of physical models can also be used such as the spatial channel model (SCM) and WINNER.

Linking the physical channel's parameters, e.g., f_{max}, angle of arrival, etc., with the parameters of the generative models is interesting topic to study. So that the generated error sequences change upon varying parameters of physical channels.

Less concentration was demonstrated for the soft bit error sequences and their modelling. More efficient and simple models could be developed in order to cover soft error sequences.

Last but not least, more applications and prediction models can be developed for error control schemes and higher layer protocols, such as protocols of the MAC layer and transport layer. The impact of the burst error statistics on designing and evaluating such schemes and protocols could also be studied.

Bibliography

[1] Wireless. (n.d.). In *Wikipedia.* Retrieved Mar. 3, 2013, from http://en.wikipedia.org/wiki/Wireless.

[2] H. Bai and M. Atiquzzaman, "Error modeling schemes for fading channels on wireless communications: a survey," *Commun. Surv. and Tutor.*, vol. 5, no. 2, pp. 2–9, Oct–Dec. 2003.

[3] V. N. Katsikis, Ed., *MATLAB - A Fundamental Tool for Scientific Computing and Engineering Applications - Volume 2.* InTech, Sep. 2012.

[4] J. G. Proakis, Ed., *Digital Communications.* 4th edition, New York: Mc Graw Hill, 2001.

[5] W. Turin, *Digital Transmission Systems: Performance Analysis and Modeling.* New York: McGraw-Hill, 1999.

[6] J. G. Proakis and M. Salehi, *Communications systems engineering.* 2nd edition, New Jersey: Prentice-Hall, 2002.

[7] W. C. Jakes, Ed., *Microwave Mobile Communications.* 2nd edition, New Jersey: IEEE Press, 1994.

[8] H. Zhu and J. Wang, "Chunk-based resource allocation in OFDMA systems-part I: joint chunk, power and bit allocation," *IEEE Trans. Wireless Commun.*, vol. 57, no. 9, pp. 2734–2744, Sep. 2009.

[9] H. Zhu and J. Wang, "Chunk-based resource allocation in OFDMA systems-part II: joint chunk, power and bit allocation," *IEEE Trans. Wireless Commun.*, vol. 60, no. 2, pp. 499–509, Feb. 2012.

[10] I. Siaud and B. Morin, "Investigation on radio propagation channel measurements at 2.2GHz and 3.5GHz for the fixed wireless access in an urban area," *Ann. Telecommun.*, vol. 54, no. 9–10, pp. 464–478, Sept–Oct. 1999.

[11] P. M. Crespo, R. M. Pelz, and J. Cosmas, "Channel error profiles for DECT," *IEE Proc.-Commun.*, vol. 141, no. 6, pp. 413–420, Dec. 1994.

[12] 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.

[13] L. Badia, N. Baldo, M. Levorato, 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.

[14] C. Pimentel and F Alajaji, "Packet-based modeling of ReedSolomon block-coded correlated fading channels via a Markov finite queue model," *IEEE Trans. Veh. Technol.*, vol. 58, no. 7, pp. 3124–3136, Sept. 2009.

[15] N. Nefedov, "Realistic discrete channel models for error control strategies evaluation," in *Proc. SBT/IEEE ITS'98*, Sao Paulo, Brazil, Aug. 1998, pp. 37–42.

[16] N. Nefedov, "Discrete channel models for evaluation of error control strategies in CDMA," *Proc. IEEE ISSSTA'98*, Sun City, South Africa, Sept. 1998, pp. 245–249.

[17] 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.

[18] 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.

[19] M. Zorzi and R. R. Rao, "Perspectives on the impact of error statistics on protocols for wireless networks," *IEEE Personal Comm.*, vol. 6, no. 10, pp. 32–40, Oct. 1999.

[20] G. Mazzini, "Queue system with smart ARQ on Markov channel for energy efficiency." in *Proc. IWCMC'05*, Hawaii, USA, June 2005, pp. 546–551.

[21] S. Tsai, "Markov characterization of the HF channel," *IEEE Trans. Commun. Technol.*, vol. 17, no. 1, pp. 24–32, Feb. 1969.

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

[23] F. Swarts and H. C. Ferreira, "Markov characterization of digital fading mobile VHF channels," *IEEE Trans. Veh. Techol.*, vol. 43, no. 4, pp. 977–985, Nov. 1994.

[24] 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.

[25] E. N. Gilbert, "Capacity of a burst-noise channel," *Bell Syst. Technol. J.*, vol. 39, pp. 1253–1265, Sept. 1960.

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

[27] S. Berkovits and E. L. Cohen,"A 3-state model for digital error distributions," *Tech. Rep. ESD-TR-67-73, The Mitre Corp.*, Bedford, MA, 1967.

[28] R. H. McCullough, "The binary regenerative channel," *Ben Syst. Technol. J.*, vol. 47, pp. 1713–1735, Oct. 1968.

[29] H. S. Wang and N. Moayeri, "Finite-state Markov channel - a useful model for radio communication channels," *IEEE Trans. Veh. Technol.*, vol. 44, pp. 163-171, Feb. 1995.

[30] Y. Yu and S. L. Miller, "A four-state Markov packet error model for the wireless physical layer," in *Proc. IEEE WCNC'07*, Hong Kong, Mar. 2007, pp. 2053–2057.

[31] 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.

[32] 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.

[33] M. Heidari, K. Pahlavan, "Markov model for dynamic behavior of ranging errors in indoor geolocation systems," *IEEE Commun. Letters*, vol. 11, no. 12, pp. 934-936, Dec. 2007.

[34] H. Steffan, "Adaptive generative radio channel models," *Proc. IEEE PIMRC'94*, Hague, Netherlands, Sept. 1994, pp. 268–273.

[35] C. Jiao, L. Schwiebert, and B. Xu, "On modeling the packet error statistics in bursty channels," in *Proc. LCN'02*, Florida, USA, Nov. 2002.

[36] H. Wang, G. Kang, K. Huang, "An advanced semi-Markov process model for performance analysis of wireless LANs," in *Proc. IEEE VTC'12-Fall*, Quebec City, Canada, Sep. 2012, pp. 1–5.

[37] P. J. Trafton, H. A. Blank, and N. F. McAllister, "Data transmission network computer-to-computer study," in *Proc. ACM 2nd Symp. on Problems in the Optimization of Data Communication Systems*, Palo Alto, CA, pp. 183–191, Oct. 1971.

[38] J-P. A. Adoul, B. D. Fritchman, and L. N. Kanal, "Characterization and modeling of real communication channels," *Tech. Rep., Dep. Elec. .Eng., Lehigh Univ.*, and Comput. Sci. Cent., Univ. Maryland, July 1970.

[39] H. A. Blank and P. J. Trafton, "A Markov error channel mode," in *Proc IEEE NTC'73*, pp. 151–158, Dec. 1973.

[40] R. T. Chien, A. H. Haddad, B. Goldberg, and E. Moyes, "An analytic error model for real channels," in *Proc IEEE ICC'72*, pp. 157–1512, Jun. 1972.

[41] Y. Wang, J. Yin, W. Fu, and D.P. Agrawal, “Performance enhancement scheme for wireless LAN under bursty Channel,” in *Proc. IEEE WCNC’07*, Hong Kong, Mar. 2007, pp. 2155–2160.

[42] 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.

[43] P. K. Varshney and A. H. Haddad, “A Markov gap model with memory for digital channels,” in *Proc IEEE ICC’75*, pp. 1524–1527, Jun. 1975.

[44] A. Semmar, M. Lecours, J. Y. Chouinard, and J. Ahern, “Characterization of error sequences in UHF digital mobile radio channels,” *IEEE Trans. Veh. Techol.*, vol. 40, no. 4, pp. 769–776, Nov. 1991.

[45] 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.

[46] L. R. Rabiner and B. H. Juang, “An introduction to hidden Markov models” *IEEE Acoust., Speech, Signal Processing Mag.*, pp. 4–16, Jan. 1986.

[47] L. R. Rabiner, “A tutorial on hidden Markov models and selected applications in speech recognition,” *Proc. of the IEEE.*, vol. 77, no. 2, pp. 257–286, Feb. 1989.

[48] M.V. Santos, E. L. Pinto, M. Grivet, “A new approach to the statistical analysis of HMM modelled bursty channels,” in *Proc. IEEE ICC’04*, Paris, France, June 2004, pp. 567–572.

[49] N. Nefedov, “Generative Markov models for discrete channel modelling,” *Proc. IEEE PIMRC’97*, Helsinki, Finland, Sept. 1997, pp. 7–11.

[50] M. Laner, P. Svoboda, and M. Rupp, “Measurement aided model design for WCDMA link error statistics ,” *Proc. ICC’11*, Kyoto, Japan, Jun. 2011.

[51] 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.

[52] O. S. Salih, C.-X. Wang, D. I. Laurenson, and Y. He, “Hidden Markov models for packet-level errors in bursty digital wireless channels,” in *Proc. LAPC’09*, Loughborough, UK, 16–17 Nov. 2009, pp. 385–388.

[53] O. S. Salih, C.-X. Wang, and D. I. Laurenson, “Soft bit error modeling for discrete wireless channels,” in *Proc. IEEE IWCMC’09*, Leipzig, Germany, 21–24 Jun. 2009, pp. 759–763.

[54] O. S. Salih, C.-X. Wang, and D. I. Laurenson, “Three-layered hidden Markov models for binary digital wireless channels,” in *Proc. IEEE ICC’09*, Dresden, Germany, 14–18 Jun. 2009.

[55] A. Beverly, K. S. Shanmugan, “Hidden Markov models for burst errors in GSM and DECT channels,” in *Proc. IEEE GLOBECOM’98*, Sydney, Australia, Nov. 1998, pp. 3692–3698.

[56] D. Gomez, R. Aguero, M. Garcia-Arranz, and L. Munoz, “On the modeling of a realistic wireless channel by means of a hidden Markov process,” in *Proc. IEEE WiMob’12*, Barcelona, Spain, Oct. 2012, pp. 397–402.

[57] W. Turin and M. M. Sondhi, “Modeling error sources in digital channels,” in *IEEE J. Select. Areas Commun.*, vol. 11, pp. 340-347, Mar. 1993.

[58] F. Jelinek, “Continuous speech recognition by statistical methods,” *Proc. IEEE*, vol. 64, pp. 532–536, Apr. 1976.

[59] Stochastic context-free grammar. (n.d.). In *Wikipedia.* Retrieved Mar. 10, 2013, from http://en.wikipedia.org/wiki/Stochastic_context_free_grammar.

[60] A. Betti, E. Costamagna, L. Favalli, P. Savazzi, “A burst-level error generation model for 2 GHz personal radio communications,” in *Proc. IEEE VTC’99-Fall*, Amsterdam, Netherlands , Sept. 1999, pp. 102–106.

[61] E. Costamagna, L. Favalli, and P. Gamba, “Multipath channel modeling with chaotic attractors,” *Proc. of the IEEE*, vol. 90, no. 5, pp. 842–859, May 2002.

[62] E. Costamagna, L. Favalli, P. Gamba, and P. Savazzi, “Block-error probabilities for mobile radio channels derived from chaos equations,” *IEEE Comm. Letters*, vol. 3, no. 3, pp. 66–68, March 1999.

[63] 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.

[64] 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.

[65] C.-X. Wang and M. Pätzold, “A novel generative model for burst error characterization in Rayleigh fading channels” in *Proc. IEEE PIMRC’03*, Beijing, China, Sept. 2003, pp. 960–964.

[66] C.-X. Wang and M. Pätzold, “A generative deterministic model for digital mobile fading channels,” *IEEE Comm. Letters*, vol. 8, no. 4, pp. 223–225, Apr. 2004.

[67] C. X. Wang and M. Pätzold, “Deterministic modeling and simulation of error sequences in digital mobile fading channels,” *Proc. IEEE ICC’04*, Paris, France, June 2004, pp. 3374-3378.

[68] C. X. Wang and M. Pätzold, “A new deterministic process based generative model for characterizing bursty error sequences,” *Proc. IEEE PIMRC’04*, Barcelona, Spain, Sep. 2004, pp. 2134–2139.

[69] C.-X. Wang and W. Xu, “Packet-level error models for digital wireless channels,” in *Proc. IEEE ICC’05*, Seoul, Korea, May 2005, pp. 2184–2189.

[70] C.-X. Wang and W. Xu, “A new class of generative models for burst-error characterization in digital wireless channels,” *IEEE Trans. Comm.*, vol. 55, no. 3, pp. 453–462, Mar. 2007.

[71] M. Pätzold, *Mobile Fading Channels.* New York: Wiley, 2002.

[72] S. O. Rice, “Mathematical analysis of random noise,” *Bell Syst. Tech. J.*, vol. 23, pp. 282–332, July 1944.

[73] S. O. Rice, “Mathematical analysis of random noise” *Bell Syst. Tech. J.*, vol. 24, pp. 46–156, Jan. 1945.

[74] G. L. Stüber, *Principles of Mobile Communications.* Boston: Kluwer Academic Publishers, 2nd edition, 2001.

[75] C.-X. Wang and M. Pätzold, and D. Yuan, “Accurate and efficient simulation of multiple uncorrelated Rayleigh fading waveforms,” *IEEE Trans. Wireless Comm.*, vol. 6, no. 3, pp. 833–839, Mar. 2007.

[76] C.-X. Wang, D. Yuan, H.-H. Chen, and W. Xu, “An improved deterministic SoS channel simulator for efficient simulation of multiple uncorrelated Rayleigh fading channels,”, *IEEE Trans. Wireless Comm.*, vol. 7, no. 9, pp. 3307–3311, Sept. 2008.

[77] 3GPP TS 45.005, “Radio transmission and reception (Rel. 4),” 2003.

[78] S. Fine, Y. Singer, and N. Tishby, “The hierarchical hidden Markov model: analysis and applications”, *Machine Learning*, vol. 32, pp. 41-62, 1998.

[79] K. Murphy and M. Paskin, “Linear time inference in hierarchical HMMs”, *NIPS’01.*

[80] H. Bui, D. Phung and S. Venkatesh, “Hierarchical hidden Markov models with general state hierarchy”, *AAAI-04.*

[81] N. Oliver, A. Garg and E. Horvitz, “Layered representations for learning and inferring office activity from multiple sensory channels”, *Computer Vision and Image Understanding*, vol. 96, pp. 163-180, 2004.

[82] Layered hidden Markov model. (n.d.). In *Wikipedia.* Retrieved Mar. 11, 2013, from http://en.wikipedia.org/wiki/Layered_hidden_Markov_model.

[83] 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.

[84] 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.

[85] G. Boggia, D. Buccarella, P. Camarda, and A. D'Alconzo, "A simple ON/OFF logarithmic model for packet-level errors in wireless channels applied to GSM," in *Proc. IEEE VTC'04-Fall*, Los Angeles, USA, Sept. 2004, pp. 4491–4495.

[86] 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.

[87] Y. He, O. S. Salih, C.-X. Wang, and D. Yuan, "Deterministic process based generative models for characterizing packet-level bursty error sequences," *Wireless Commun. and Mobile Computing*, 2013, DOI: 10.1002/wcm.2356.

[88] M. Pätzold and D. Kim,"Test procedures and performance assessment of mobile fading channel simulators, in *Proc. IEEE VTC04-Spring*, Milan, Italy, May 2004, pp. 254–260.

[89] C. Mehlführer, 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. [Online]. Available: http://publik.tuwien.ac.at/files/PubDat_175708.pdf

[90] C. Mehlführer, J. C. Ikuno, M. Simko, S. Schwarz, M. Wrulich, and M. Rupp, "The Vienna LTE simulators - enabling reproducibility in wireless communications research," *EURASIP Journal on Advances in Signal Processing*, vol. 2011, no. 1, pp. 1–13, 2011. [Online]. Available: http://publik.tuwien.ac.at/files/PubDat_199104.pdf

[91] LTE Downlink Link Level Simulator. (n.d.). In *Vienna University of Technology*. Retrieved July. 07, 2013, from http://www.nt.tuwien.ac.at/about-us/staff/josep-colom-ikuno/lte-downlink-link-level-simulator/

[92] M. Dottling, J. Michel, and B. Raaf, "Hybrid ARQ and adaptive modulation and coding schemes for high speed downlink packet access," *Proc. IEEE PIMRC'02*, Lisbon, Portugal, Sep. 2002, pp. 1073–1077.

[93] B. Kian Chung, A. Doufexi, and S. Armour, "Performance evaluation of hybrid ARQ schemes of 3GPP LTE OFDMA system," *Proc. IEEE PIMRC'07*, Athens, Greece, Sep. 2007, pp. 1–5.

[94] J. C. Ikuno, M. Wrulich, and M. Rupp, "Performance and modeling of LTE H-ARQ," *Proc. WSA'09*, Berlin, Germany, Sep. 2007.

[95] Y. W. Blankenship, P.J. Sartori, B.K. Classon, V. Desai, and K.L. Baum, "Link error prediction methods for multicarrier systems," *Proc. VTC'04-Fall*, Los Angeles, USA, Sep. 2004, pp. 4175–4179.

[96] B. Classon, P. Sartori, Y. Blankenship, K. Baum, R. Love, and Y. Sun, "Efficient OFDM-HARQ system evaluation using a recursive EESM link error prediction," *Proc. WCNC'06*, Las Vegas, NV, USA, Apr. 2006, pp. 1860–1865.

[97] J. Kim, A. Ashikhmin, A. V. Wijngaarden, E. Soljanin, and N. Gopalakrishnan, "On efficient link error prediction based on convex metrics," *Proc. VTC'04-Fall*, Los Angeles, USA, Sep. 2004, pp. 4190–4194.

[98] J. C. Ikuno, C. Mehlfuhrer, and M. Rupp, "A novel link error prediction model for OFDM systems with HARQ ," *Proc. ICC'11*, Kyoto, Japan, Jun. 2011, pp. 1–5.

[99] A. Das and A. Sampath, "Link error prediction for wireless system simulations," *Proc. WCNC'04*, Atlanta, Georgia, USA, Mar. 2004, pp. 507–512 .

[100] W. Karner, O. Nemethova, and M. Rupp, "Link error prediction in wireless communication systems with quality based power control," *Proc. ICC'07*, Glasgow, UK, Jun. 2007, pp. 5076–5081.

[101] M. Lampe, H. Rohling, and J. Eichinger, "PER prediction for link adaptation in OFDM systems," *7th International OFDM Workshop*, Hamburg, Germany, Sept. 2002.

[102] S. Simoens, S. Rouquette-Leveil, P. Sartori, Y. Blankenship, and B. Classon, "Error prediction for adaptive modulation and coding in multiple-antenna," Tech. Report, Jan. 2005. [Online]. available at: http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.61.5548.

[103] O. S. Salih, C.-X. Wang, R. Mesleh, X. Ge, and D. Yuan, "Predicting burst error statistics of digital wireless systems with HARQ," in *Proc. IEEE IWCMC'13*, Cagliari, Sardinia, Italy, 1–5 Jul. 2013, pp. 276–281.