PUBLIC

IST R&D. FP6-Priority 2.
SPECIFIC TARGETED RESEARCH PROJECT
Project Deliverable

SUIT Doc Number	SUIT_209
Project Number	IST-4-028042
Project Acronym+Title	SUIT- Scalable, Ultra-fast and Interoperable Interactive Television
Deliverable Nature	Report
Deliverable Number	D2.4
Contractual Delivery Date	30 Nov 2006
Actual Delivery Date	21 Dec 2006
Title of Deliverable	WLAN Modelling
Contributing Workpackage	WP2
Project Starting Date; Duration	01/02/2006; 27 months
Dissemination Level	PU
Author(s)	C. H. Liew (UniS), S. Worall (UniS), J. C. Plaza (UPM), J. Cabrera (UPM), F. Jaureguizar (UPM), N. García (UPM)

Abstract

This report features a discussion of the simulation results and an analysis of WLAN. Furthermore, the report outlines the developed simulator, the error pattern trace generation method and the error pattern format. In addition to the simulator generated trace, an error pattern trace file collected from a real network is also presented and documented. Based on the real observations, packet-level channel models have been developed. These models have been integrated into a packet-level simulator and allow the inclusion of the channel behaviour in the development of rate control strategies. Either error pattern trace file or the packet-level channel models can be used in WP3 for video performances study.

Keyword list: WLAN, IEEE 802.11g, Wi-Fi, discrete packet level channel model, error pattern file

WLAN Modelling

SUIT_209

21 December 2006

Table of Contents

1 Introduction to WLAN Modelling

This section outlines the basic baseband model of IEEE802.11g for simulator development. Using the simulator, a set of physical link level graphs has been produced for different sets of simulation parameters. Analysis and observations concerning the graphs have been provided in section 3. The graphs will be used as a basis for cross layer optimization of video transmission in WP3. Together with the graphs, a set of packet error traces is generated using the developed WLAN simulator. In addition to the simulator generated trace, a packet error trace from a real WLAN network has also been collected by the SUIT partners. Based on these real observations, packet-level channel models have been developed which have been integrated in a packet-level simulator. Both the graphs and the packet error trace enable optimal selection of video coding parameters for video stream transmission. This is also important for the design of intelligent adaptation schemes for the WLAN gateway within the SUIT project. The usage scenario of the WLAN gateway can be seen in Figure 1 [1] where the gateway receives MDC streams from DVB and WIMAX systems. The MDC streams are combined in the gateway and the output bit rate is scaled accordingly to adapt to the local channel variation of WLAN network. The robustness of the video adaptation schemes and rate control strategies can be tested using packet error traces before the final system deployment. Moreover, the characterization of the channel behaviour can be included as a key element in the rate control strategies through the developed channel models.

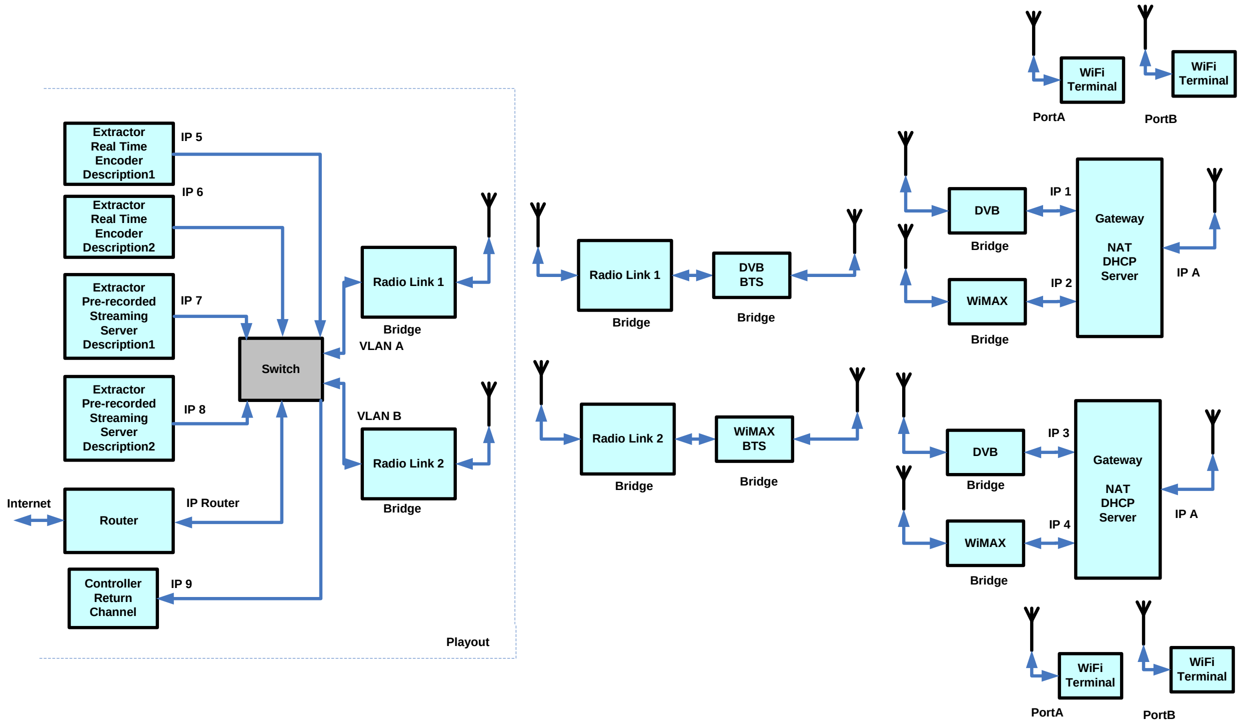

Figure 1 SUIT system for home network scenario

In Section 2, the system specifications and baseband system model is outlined. WLAN simulator software in C++ is developed according to the system specification and model. Results generated using the simulator software are then presented in Section 3. Section 4 outlines the packet error trace generation format and parameters. In addition, the functions of each module in the simulator software are described. Section 5 proposes three discrete packet level channel models and evaluates their performance under diverse transmission conditions. These models are trained with data collected from a real environment, and they are able to generate simulated packet level traces of an arbitrary length Finally the deliverable is concluded in section 6.

2 WLAN System Descriptions

This section describes the WLAN baseband simulation model used in the SUIT project. The section first outlines the system specifications and then gives an overview of the implemented model. The simulator validation is then presented by comparing the simulation results to the results in the literature.

2.1 WLAN System Specifications and Baseband System Model

2.1.1 System Specifications

The summary of the system parameters used for WLAN IEEE802.11g in the project is presented in Table 1. The number of parameters in IEEE802.11g is not numerous and thus less complicated when compared to other systems such as DVB-T and WIMAX. The implementation of the WLAN system model described in the following sections is based on the specification in Table 1. The associated Orthogonal Frequency Division Multiplexing (OFDM) parameters, which the simulator is based on, are presented in Table 2.

Table 1 System specification of IEEE 802.11g

System Parameter	Description
Carrier Modulation	OFDM
FFT Size	64
Carrier Frequency	2.4 GHz
Sampling rate	20 MHz
Channel Coding	Punctured Convolutional Coding
Modulation	BPSK, QPSK, 16QAM, 64QAM

Table 2 OFDM parameters

OFDM Parameter	Value
Sampling rate	20 MHz
Useful symbol duration	3.2 µs
Guard interval duration	0.8 µs
Total symbol duration	4.0 µs
FFT size	64
Number data subcarriers	48
Number of pilot subcarriers	4
Subcarrier Spacing	0.3125 MHz
Total bandwidth	16.875 MHz

The combination of channel coding and modulation schemes in Table 1 produces several transmission modes with different data rate. A particular combination of channel coding and modulation scheme is also called the Modulation and Coding Scheme (MCS). The MCS modes are shown in Table 3.

Table 3 Modulation and Coding Schemes (MCS) in IEEE802.11g

MCS	Modulation	Bit Rate (Mbps)	Coded bits per OFDM symbol	Data bits per OFDM symbol
0	BPSK	6	48	24
1	BPSK	9	48	36
2	QPSK	12	96	48
3	QPSK	18	96	72
4	16-QAM	24	192	96
5	16-QAM	36	192	144
6	64-QAM	48	288	192
7	64-QAM	54	288	216

Note that the bit rate in Table 3 refers to the physical layer bit rate. The actual data throughput for any given MCS mode will be lower due to protocol layer overhead.

2.1.2 Overview of Implemented WLAN System Model

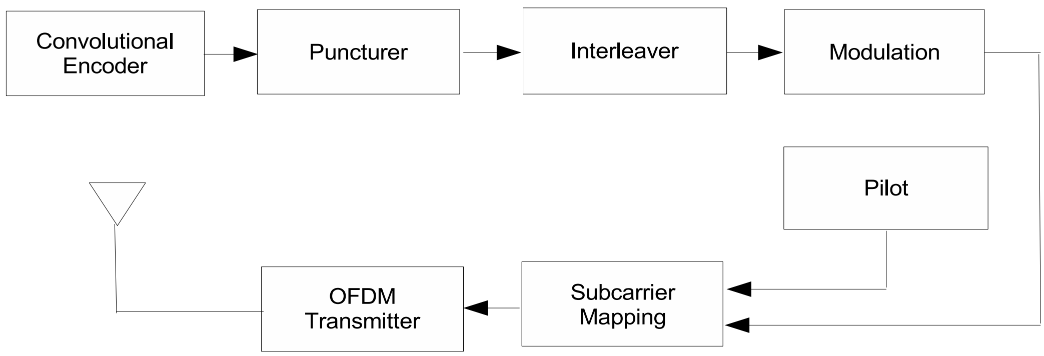

Figure 2 Baseband Model of IEEE 802.11g

The implemented baseband system model of IEEE 802.11g is shown above. The model is based on the Extended Rate Physical - Orthogonal Frequency Division Multiplexing (ERP-OFDM) mode which provides data rates ranging from 6Mbps to 54Mbps. Elements of the system model, as shown in Figure 2, are:

- Convolutional Coding
- Puncturing
- Interleaving
- Modulation
- Pilot and data symbols mapping onto OFDM subcarriers
- OFDM Modulation

Each of the above elements will be discussed briefly in the following sections. The C++ software implementation of the baseband model makes use of the IT++ communication signal processing library [2].

2.1.2.1 Convolutional Coding

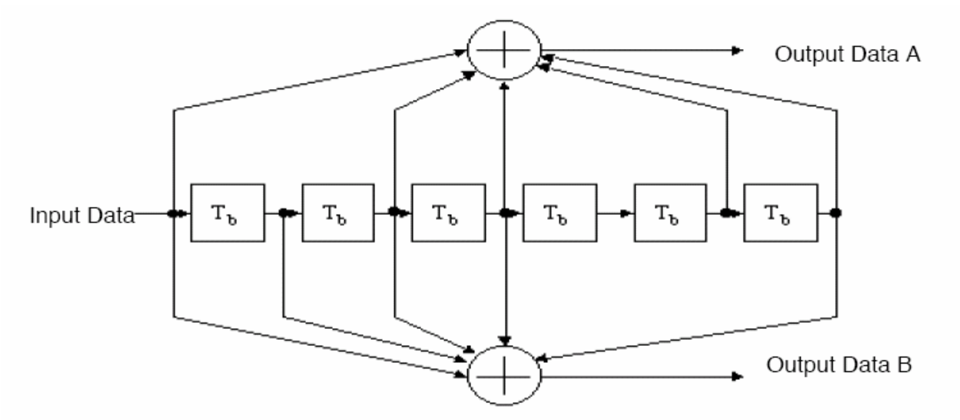

Figure 3 Convolutional encoder with constraint length 7

The Convolutional Coding in the standard uses the industry standard generator polynomials, $g_0 = 133_8$ and $g_1 = 171_8$, of rate R = 1/2 as shown in Figure 3. Higher data rates can be achieved by puncturing. The puncturing operation is described in the next section. Soft-decision Viterbi Algorithm is adopted for decoding.

2.1.2.2 Puncturing

Puncturing is a procedure to omit some of the encoded bits from the channel coding module to increase the transmission data rate. The puncturing of output encoded bits “A” and “B” in Figure 3 are shown in the following figures:

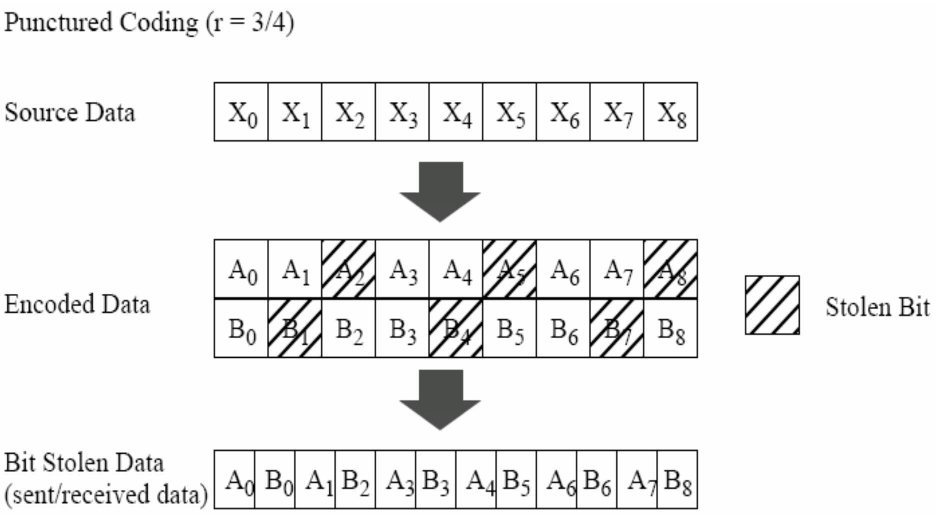

Figure 4 Puncturing of encoded bits for code rate of 3/4

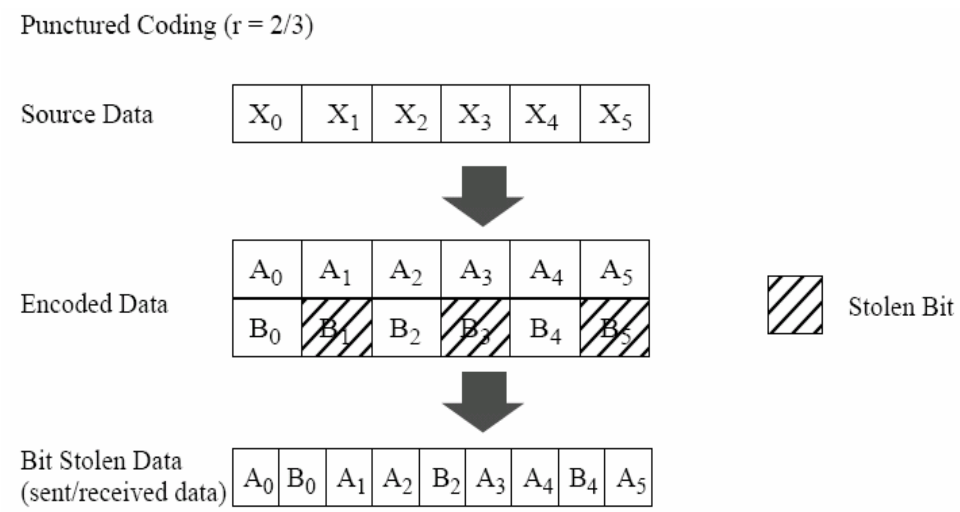

Figure 5 Puncturing of encoded bits for code rate of 2/3

The “Stolen Bits” in above figures are the omitted bits. At the receiver side, the zero-valued dummy bits are inserted at the “Stolen Bit” location before the channel decoding operation.

2.1.2.3 Interleaving

The interleaving operation ensures that the adjacent coded bits are mapped onto non-adjacent OFDM subcarriers and at the same time maps coded bits alternately onto less and more significant bits of the modulation constellation. The aim of this technique is to increase the robustness and avoid long runs of low reliability bits. The operation is divided into two permutation steps. The first permutation equation is:

$$i = (N_{CBPS}/16)(k \bmod 16) + \lfloor k/16 \rfloor \qquad k = 0,\ldots,N_{CBPS}-1 \tag{1}$$

where i is the index after first permutation of index k. k is the index of coded bits before first permutation. N_{CBPS} is the number of bits in a single OFDM symbol. The second permutation mapping index i to j is:

$$j = s \times \lfloor i/s \rfloor + (i + N_{CBPS} - \lfloor 16 \times i / N_{CBPS} \rfloor) \bmod s \qquad i = 0,\ldots,N_{CBPS}-1 \tag{2}$$

where $s = \max(N_{BPSC}/2,1)$ with N_{BPSC} being the number of bits per modulation symbol.

2.1.2.4 Modulation Mapping

OFDM subcarriers are modulated using BSPK, QPSK, 16-QAM and 64-QAM modulation. The encoded and interleaved serial input data is divided into groups of N_{BPSC}, i.e. 1, 2, 4 or 6, bits which will then be converted into a complex number, $I + Q \times j$, representing either BPSK, QPSK, 16-QAM, and 64 QAM constellation point. The mappings for BPSK, QPSK, 16-QAM and 64-QAM are shown in Figure 6. Finally, the resulting complex value is normalized by multiplying the normalization factor, K_{MOD}, specified in Table 4.

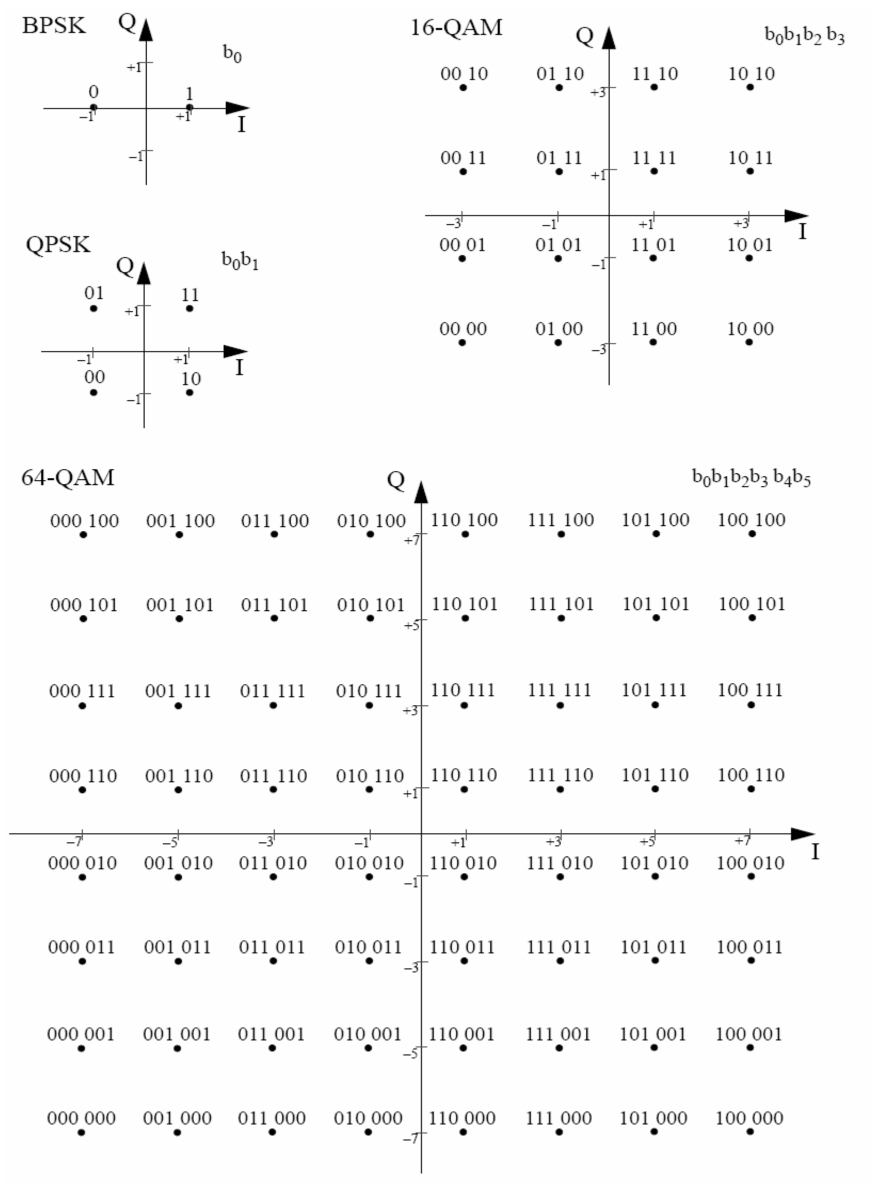

Figure 6 Mapping of serial input bits into complex value

Table 4 Modulation-dependent normalization factor

Modulation	K_{MOD}
BPSK	1
QPSK	$1/\sqrt{2}$
16-QAM	$1/\sqrt{10}$
64-QAM	$1/\sqrt{42}$

2.1.2.5 Pilot and Data Symbol Mapping onto OFDM Subcarriers and OFDM Modulation

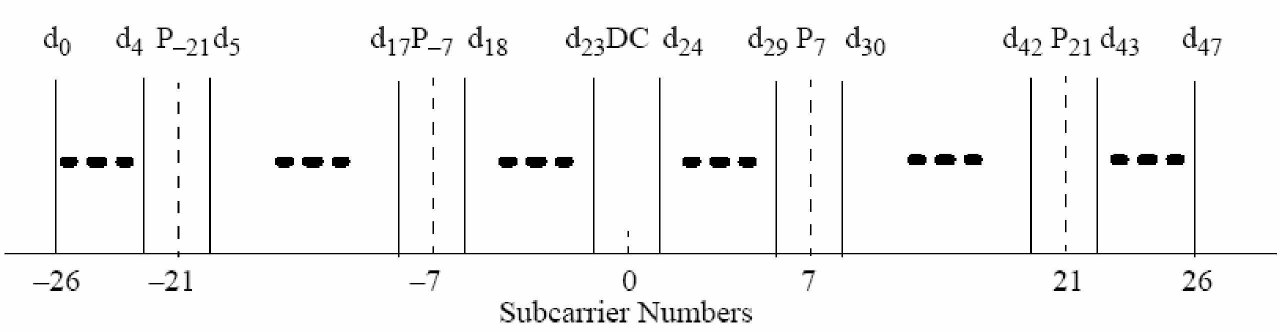

Figure 7 Arranging pilot and data symbols

Pilot symbols are inserted into OFDM symbol to allow robust signal detection against frequency offsets and phase noise. The generation of the pilot sequence is outlined in [3]. Pilot symbols and data symbols are arranged in the order shown in Figure 7. d_0 to d_{47} are the sequential data symbols while P_{-21}, P_{-7}, P_7 and P_{21} are the pilot symbols. The pilot and data symbols are then loaded onto the OFDM subcarriers following the subcarrier numbering shown in Figure 8. The Inverse Fast Fourier Transform (IFFT) is performed on all subcarriers to compute the time domain symbol with 64 output samples. Finally, cyclic prefix insertion is performed by copying 16 samples from index 48th to 63th of the time domain symbol and appends them to the beginning of the symbol. Thus the total number of output samples of a OFDM symbol is 80.

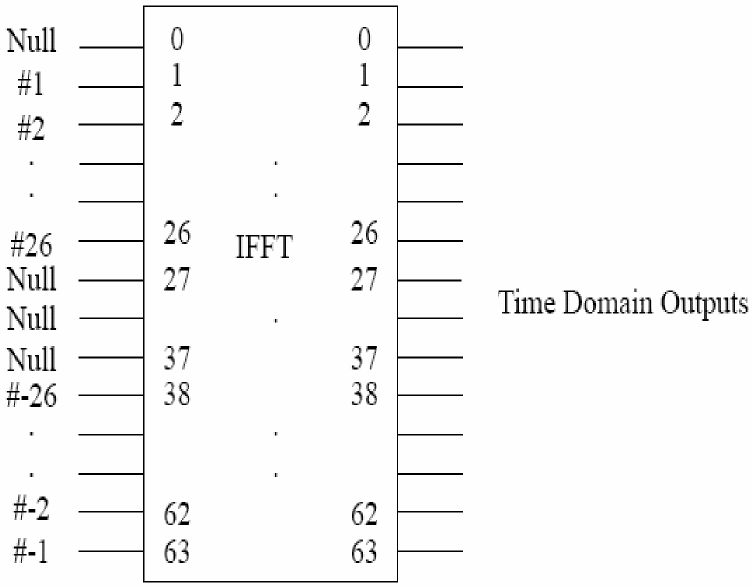

Figure 8 Loading pilot and data symbols onto subcarriers followed by OFDM modulation

2.1.2.6 Channel Estimation and Equalizer

GI	T1	T2

Figure 9 Long training symbol for channel estimation

Two long training symbols are defined in IEEE 802.11g standard for frequency offset and channel estimation purposes. The training symbols, T_1 and T_2, are placed together as shown in Figure 9. The guard interval, GI, is twice the duration of the normal OFDM data symbol guard interval to ensure robustness against inter-symbol interference. The channel can be estimated by averaging the frequency domain value of T_1 and T_2 (after the FFT operation of OFDM receiver). The equalizer used in the simulation is the zero forcing equalizer [4].

2.1.3 Simulator Validation

The simulator developed for the SUIT project is compared to the results published in [5]. In the simulation, HiperLAN/2 Channel Model A [6] has been used. The Channel Impulse Response (CIR) of the channel model was normalized during the simulation. The packet size used for generating the graph was 512 bytes. The packet error rate of SUIT WLAN simulator is shown in Figure 10. It can be observed that the curves generated using the SUIT WLAN simulator correspond closely to those found in the existing literature. This demonstrates the correctness of the simulator.

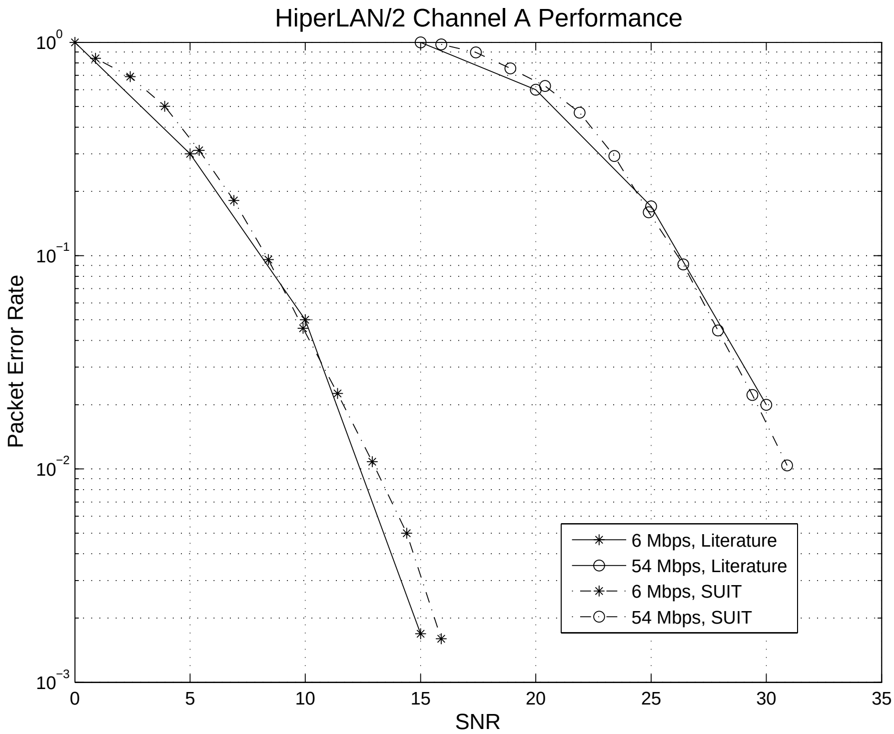

Figure 10 Comparison of the SUIT WLAN simulator to results in the literature

3 Physical Layer Simulation Results and Analysis

This section discusses the results of physical layer simulation. Firstly, the simulation parameters are given. Secondly, the results for different simulation parameters are listed and finally some brief discussions of the results are presented.

3.1 Simulation Parameters

The simulation parameters used for the IEEE802.11g study are summarized in Table 5.

Table 5 Summary of simulation parameters

Parameter	Value
MCS Mode	0 - 7
Packet Size (bytes)	256, 512, 1024
Channel Model	HiperLAN/2 Channel Model A – Channel Model E [6]

For convenience, the characteristics of the adopted channel models are summarized from [6] in Table 6. A comprehensive description of channel models used can be found in [6].

Table 6 Channel model characteristics

Channel Model	RMS Delay Spread (ns)	Characteristics	Environment
A	50	Rayleigh	Small office NLOS
B	100	Rayleigh	Medium office NLOS
C	150	Rayleigh	Large office NLOS
D	140	Rayleigh	Large office LOS
E	250	Rayleigh	Outdoor NLOS

Note that the CIR of the channel model was normalized during the simulation such that the average input power is equal to the average output power from the channel model.

3.2 Simulation Results and Analysis

The simulated BER curves are shown in Figure 11 to Figure 15. These curves can be used as a basis for optimization of video transmission over WLAN depending on the deployment environment. One of the observation from the graphs is that the degradation in performance of mode MCS1 (BPSK-3/4) is large when compared mode MCS2 (QPSK-1/2). This is because the punctured convolutional code does not cope well with frequency selective fading. Also, in general, it can be seen that the BER performance improves as the delay spread increases. This is evident from the fact that the post- receiver SNR or Es/No progressively reduces for a same BER requirement. The observation is consistent with the observation in [7]. This is because the increasing delay spread decorrelates the subband fading, thereby achieving good frequency diversity [7]. However, significant BER error floor is observed in the case of Channel E for MCS Mode 5 to Mode 7. This is due to the effect of inter-symbol interference (ISI) when deploying WLAN in a large open space or an outdoor environment, where the delay of echo signals are comparable to the symbol period. Thus MCS Mode 5 to Mode 7 are never useful in such environments, as using these modes drastically reduces the throughput.

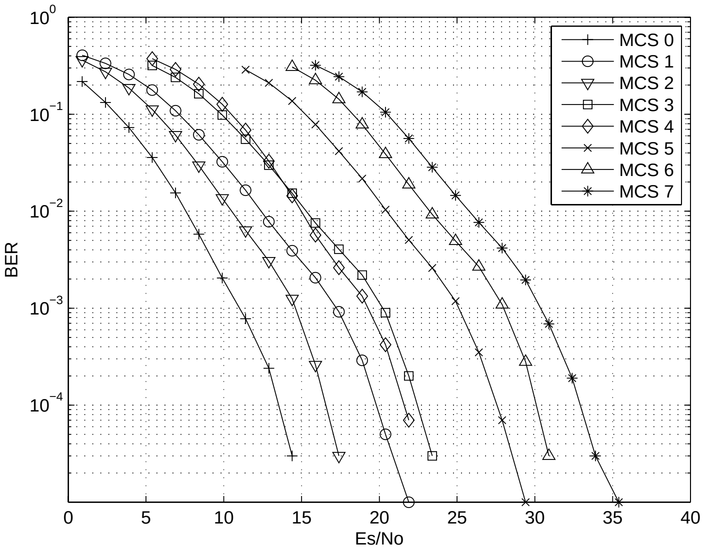

Figure 11 BER of WLAN in Channel A

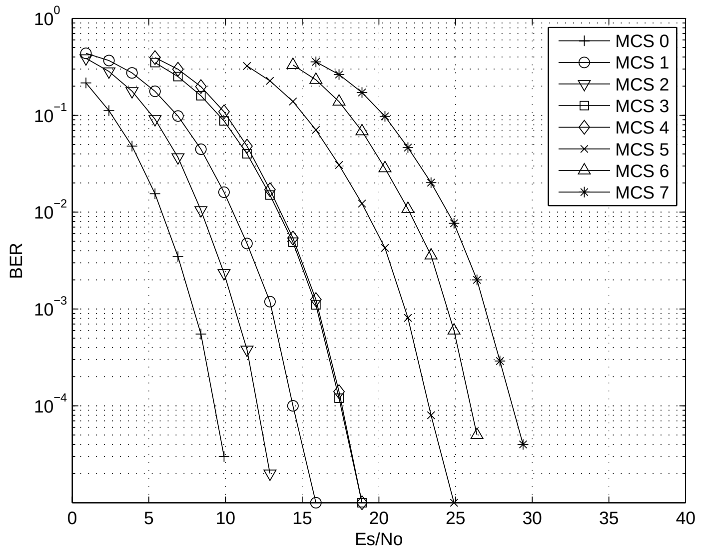

Figure 12 BER of WLAN in Channel B

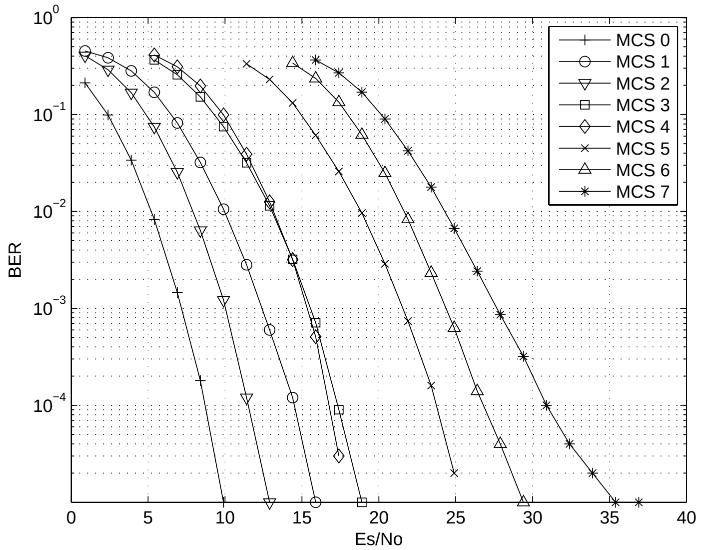

Figure 13 BER of WLAN in Channel C

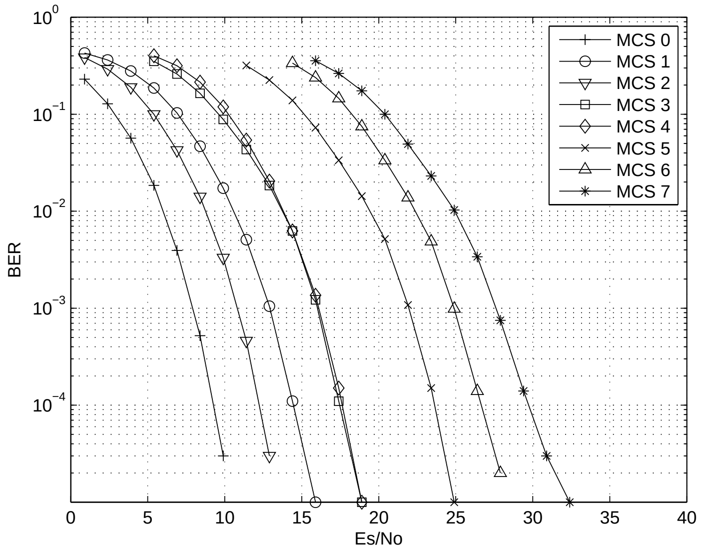

Figure 14 BER of WLAN in Channel D

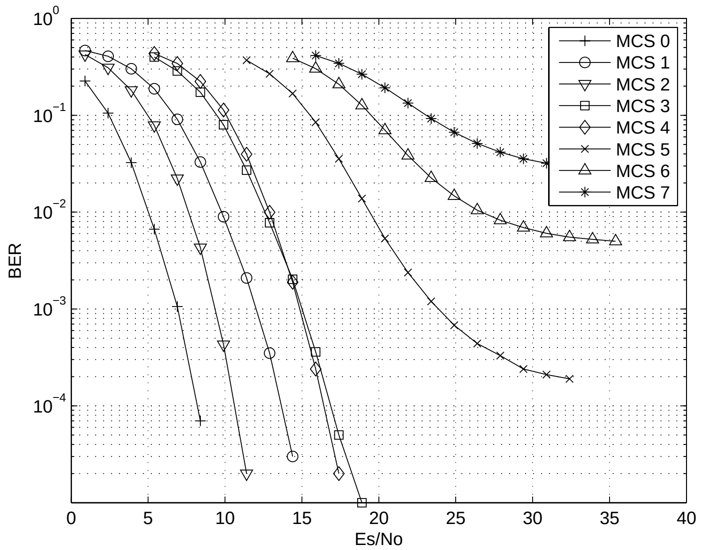

Figure 15 BER of WLAN in Channel E

4 Error Pattern Files Generation

This section discusses the data flow over the WLAN protocol layers. Following the discussions, the simulator software of the protocol layer coded in C++ (using the IT++ signal processing library) is outlined and the functions of each module are provided. Finally, the parameters used for trace generation and the trace format are described.

4.1 Data Flows over the Protocol Layer

30 Bytes	3 Bytes	5 Bytes		4 Bytes
MAC Header	LLC Header	SNAP Header	IP Data	FCS

Figure 16 IEEE 802.11g data link layer frame format

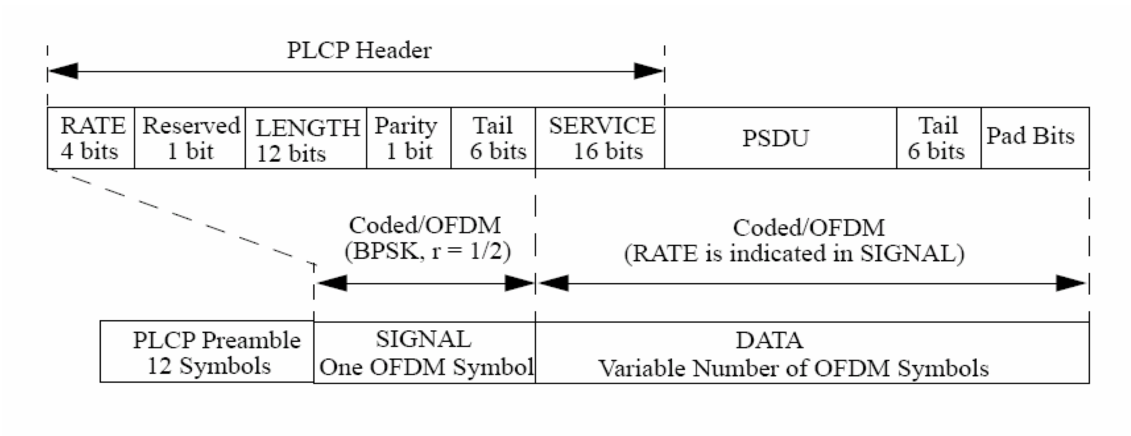

Figure 17 IEEE 802.11g physical layer frame format

It is not the aim of this section to discuss every field of the protocol layer in details. The reader may refer to [3;8] for a detailed treatment if required. It is sufficient to show how the IP packet is encapsulated in the IEEE 802.11g OSI data link layer in Figure 16. Basically the Link Layer Control (LLC) and Sub- Network Access Protocol (SNAP) provide an access interface to network layer protocols to use the service of the MAC sublayer in the data link layer. LLC and SNAP are defined in such a way that the MAC sublayer is independent from the type of network layer protocols used, e.g. ATM or IP. As for the MAC sublayer, it is used to control user access to the wireless transmission medium. The Frame Check Sequence (FCS) is a number of parity bits, used for error detection. The data link layer frame in Figure 16 is then passed, as a Protocol Service Data Unit (PSDU), to the physical layer for transmission, as shown in Figure 17. The physical layer frame is composed of Physical Layer Convergence Procedures (PLCP) preamble, PLCP header, PSDU, tail bits and pad bits. PLCP preamble is used for time and frequency acquisition, as well as for channel estimation procedures. The PLCP header specifies the format of the frame, i.e. MCS mode, length of frame and etc. As for the tail bits, they are used to flush the memory of convolutional encoder. Finally the padding bits are used to pad the frame to ensure that it fits exactly into an integral number of OFDM symbols.

4.2 Simulator Software Descriptions

The simulation software developed in the project is listed below. The function of each module is described in the Table 7.

Table 7 Simulation software files description

Files	Descriptions
psdu.h psdu.cc	Randomly generates IP data, MAC header, and performs CRC. Extracts IP data at the receiver for error calculation.
scrambler.h scrambler.cc	Scramble and descramble the data before channel coding
wlanconv.h wlanconv.cc	Perform convolutional coding and decoding and puncturing operation
interleaver.h interleaver.cc	Perform interleaving and de-interleaving
modem.h modem.cc	Performs BPSK, QPSK, 16QAM and 64QAM mapping
pilot.h pilot.cc	Generates pilots for insertion into OFDM symbol for frequency offset correction etc
freqmux.h freqmux.cc	Multiplexing data and pilot into OFDM subcarriers location
wlanofdm.h wlanofdm.cc transceiver.h transceiver.cc	Perform OFDM operation, which includes zero padding, IFFT / FFT, and cyclic prefix insertion / removal
chanmodel.h chanmodel.cc	Initialize TDL channel model of IT++ and set the parameters according channel profile i.e. HiperLAN/2 Channel Models
plcp.h plcp.cc	Generates training symbols for synchronization, frequency offset and channel equalization purposes. Performs channel estimation.
simparam.h simparam.cc	Defines all the simulator parameters
wlan_multipath_bran.cc wlan_multipath_bran_trace.cc	Main simulator file integrating individual modules above for estimating BER and generating error pattern files.

4.3 Pre- Generated Error Trace Format

The developed simulator has been used to generate packet error trace. It should be noted that the packet error trace generation assumes that there is no acknowledgement on receiving packets. This is in fact the case when IEEE802.11g operates in multicasting and broadcasting mode [9] which is the target transmission scenario for SUIT project i.e. broadcasting of TV to portable clients. During the trace generation, frames are formatted according to Figure 16 and Figure 17. The length of all the trace generated is 60 seconds. The number of packets and the packet size used are specified in Table 8. Note that the number of packets can be calculated by dividing length of trace to the packet duration. Packet duration in Table 8 is calculated as follows:

$$\begin{aligned}\text{Packet Duration } (\mu s) &= \text{Preamble Duration} + \text{Signal Symbol Duration} + \text{DATA Field duration} \\ &= 16 + 4 + (\text{DATA Bits}/(\text{Num of Bits per OFDM Symbol in MCS Mode})) \times 4\end{aligned} \quad (3)$$

where preamble duration and signal duration can be obtained as 16μs and 4μs respectively from Figure 110 in [3]. The DATA Bits (refer to Figure 17) is computed as follows:

$$\begin{aligned}\text{DATA Bits} &= \text{MAC Header Bits} + \text{LLC Header Bits} + \text{SNAP Header Bits} + \\ &\quad \text{IP Packet Bits} + \text{CRC Bits} + \text{Padding Bits} + \text{Service Bits} + \text{Tail Bits} \\ &= (30 + 3 + 5 + \text{IP Packet Bytes} + 4) \times 8 + \text{Padding} + 16 + 6\end{aligned} \quad (4)$$

Table 8 Parameters used for packet error trace generation

MCS Mode	Number of Bits per OFDM Symbol	Packet Size (bytes)	Packet Duration (us)	Length of Trace (s)	Approximate Number of Packets
0	24	256	424	60	141509
	24	512	764	60	78534
	24	1024	1448	60	41436
1	36	256	288	60	208333
	36	512	516	60	116279
	36	1024	972	60	61728
2	48	256	224	60	267857
	48	512	392	60	153061
	48	1024	736	60	81521
3	72	256	156	60	384615
	72	512	268	60	223880
	72	1024	496	60	120967
4	96	256	124	60	483870
	96	512	208	60	288461
	96	1024	380	60	157894
5	144	256	88	60	681818
	144	512	144	60	416666
	144	1024	260	60	230769
6	192	256	72	60	833333
	192	512	116	60	517241
	192	1024	200	60	300000
7	216	256	68	60	882352
	216	512	104	60	576923
	216	1024	180	60	333333

HiperLAN/2 Channel Model A has been used for packet error trace generation for a typical office environment. Nevertheless the packet error trace for other scenarios or channel models can be generated easily with the SUIT simulator made available to the consortium. For each MCS mode, 5 – 7 data points, each representing different SNR/BER level, are generated. In the generated trace, symbol “1” refers to a packet error while symbol “0” means that there is no error. The traces are named using the following context, “trace.mcsX.Y” where mcsX refers to the MCS mode X while Y refers to the SNR value. A companion file “ber.X” that is submitted together with the traces contain the corresponding BER and PER values of trace generated with MCS mode X and SNR Y.

In Table 3, the IEEE 802.11g standard have a quoted bit rate of up to 54Mbps. This is in fact the physical layer bit rate and the actual throughput is lower due to overheads at the MAC and physical layer. It is more efficient to transmit a larger packet as the protocol overhead is fixed. To match the video bit rate to a particular MCS mode, the bit rate can be calculated by dividing the target packet size by the packet duration, calculated using Equation 3.

5 Discrete WLAN Channel Modelling

The aim of activity 2.4 is to find accurate models for the Wi-Fi transmission system, including network, link and physical layers. This section focuses on the development of discrete-time channel models at packet level, being aware of WLAN stochastic complex behaviour (bursty random errors, fast variations…).

This information will be used in WP3 to optimize the performance of the system according to the network and service characteristics. More specifically, the channel behaviour can be included in the development of appropriate rate control strategies through these discrete channel models.

5.1 State-Based models

Discrete channel models usually try to capture the behaviour of the transmission medium though the definition of a set of states and possible transitions among them. These states are generally linked to a bit/packet error probability, and can have some implicit physical meaning.

Markov chains [10] and Hidden Markov Models (HMMs) [11], both of them state-based, are powerful statistical structures capable of simulating a great variety of stochastic processes. Many of the most popular models are particularization of these tools.

Markov chains are discrete-time stochastic processes with the Markov property. A Markov chain characterizes a system that goes through a set of different states, according to fixed transition probabilities. Each one of these states is linked to an observable physical event (represented with a symbol), such as the successful transmission of a single packet, or an interval of SNR values. This way, it is straightforward to obtain a sequence of states from the sequence of observed symbols, and it is possible to know (certainly, at any moment) in which state the process is.

A Markov chain is completely determined by:

- A set of N different states: $\{S_1, S_2\ldots, S_N\}$
- The transition probability matrix, among those states: $A=\{a_{ij}\}$, from S_i to S_j, *where:*

$$a_{ij} \leq 0$$

$$\sum_{j=1}^{N} a_{ij} = 1$$

- A vector of initial state occupation probabilities: $\pi_j=P[x(t)=S_j]$

Some of the most used discrete channel models follow this structure. However, the deterministic correspondence between a physical event and a concrete state in the chain may be too restrictive for some applications.

On the other hand, Hidden Markov Models (HMMs) aims at modelling a system with the Markov property whose states are not observable. Thus, observations are modelled as probability functions in every state. This way, a HMM is twice a stochastic process, since it is not possible to fully determine at which state the model is, but only with certain probability. There is no one-to-one link between symbol and state, and different state sequences can drive to the same observation sequence.

Discrete HMMs are characterized by the following parameters:

- The number of states in the model, N
- The number of symbols each state can emit, M
- The probability transition matrix, between the different states, $A=\{a_{ij}\}$
- The emission probability matrix, for each state and every symbol, $B=\{b_{ik}\}$

- The vector of initial state occupation probabilities, $\pi_j=P[x(t)=S_j]$

A common notation considers an HMM defined by λ, where $\lambda = (A, B, \pi)$.

Training the parameters of a HMM λ involves obtaining the distributions that maximize the conditional probability $P(O(t)/\lambda)$ for a known sequence of observations, $O(t)$. This duty is generally accomplished by Baum-Welch algorithm.

As aforesaid, it is not possible to know certainly in which one of the different states the process is at every moment. However, it is feasible to estimate the most probable sequence of states by means of the Viterbi algorithm.

5.2 Proposed discrete channel models

In order to obtain a good estimation of Wi-Fi channel behaviour, some research has been made on three different discrete channel models. They will be now explained, in growing grade of complexity.

5.2.1 Gilbert Model

For this activity, a simplified Gilbert model [12] has been used as a "benchmark". This scheme is widely employed for channel modelling tasks, and can be seen as a two state particularization of the Markov chain model. It consists of a good and a bad state, for which the result of the packet transmission is either successful or erroneous. Hence, these states are linked to an observable physical event (success or not in the reception of a symbol). Figure 18 represents the two states of a Gilbert model together with its transition probabilities.

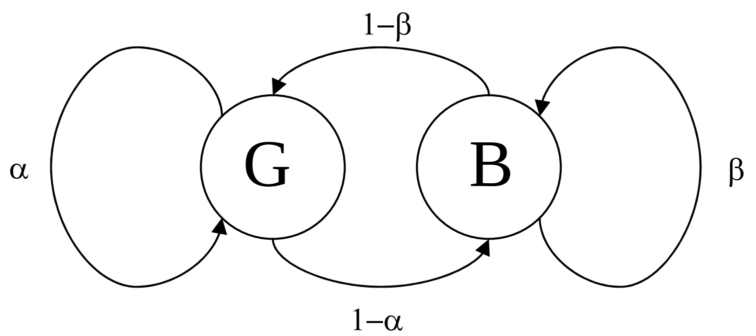

Figure 18 Gilbert model diagram

Since we are assuming that the probability of a correct packet reception in the good state is P_g=1 and in the bad state is P_b=0, the probability mass functions (pmf) of the length of error bursts (X) and error-free bursts (Y) follow a geometric distribution as follows:

$$P[X=n]=\alpha^{n-1}(1-\alpha)$$

$$P[Y=m]=\beta^{m-1}(1-\beta)$$

Model parameters can be directly computed from the mean values of the error bursts and error-free bursts as follows:

$$\alpha=1-1/E[X]$$

$$\beta=1-1/E[Y]$$

5.2.2 Class Partitioned Markov Chain model

This approach consists of the combination of several 2-states Markov models by means of a N-state Markov chain. The objective of this model is to extend the Gilbert model to be able to capture a more complex behaviour.

In this scheme, higher level information is considered in the modelling: the packet error rate (PER). The N-state Markov chain models the behaviour of the transmission channel in terms of packet error rate (PER). The behaviour of the channel is classified in N-classes according to observed PER along a cluster of observed packet transmissions of length *L*. Each state then represents a class, and the system moves from one class to the next. Within each class, a simple Gilbert model characterizes the transmission of *L* packets. Figure 19 shows an example of a 3-state model.

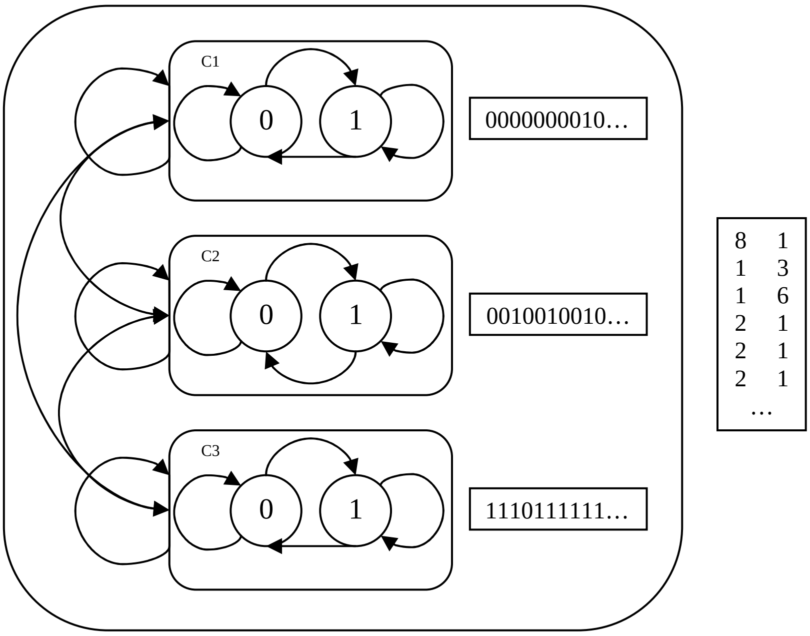

Figure 19. 3-states Class Partitioned Markov Chain model

For the training of the model implemented in this activity, firstly input data are divided into L-packet clusters. These clusters are split into N classes according to their PER. This classification is done using Max-Lloyd algorithm [13]. For the calculation of transition probabilities between PER classes, Baum-Welch algorithm [14] is applied.

In this model, there are two parameters, *N* and *L*, which may affect the accuracy of the models. Therefore, they should be tuned according to the specific scenario conditions.

5.2.3 Hidden Markov Models

Hidden Markov Models [15] are an interesting approach to the objectives of this activity, for their flexibility and accuracy in the extraction of the statistical characteristics of one-dimensional processes. One of their most powerful advantages is that, though their states are not necessarily linked to physical events, they could carry inherent information about them. Figure 20 shows an example of an HMM.

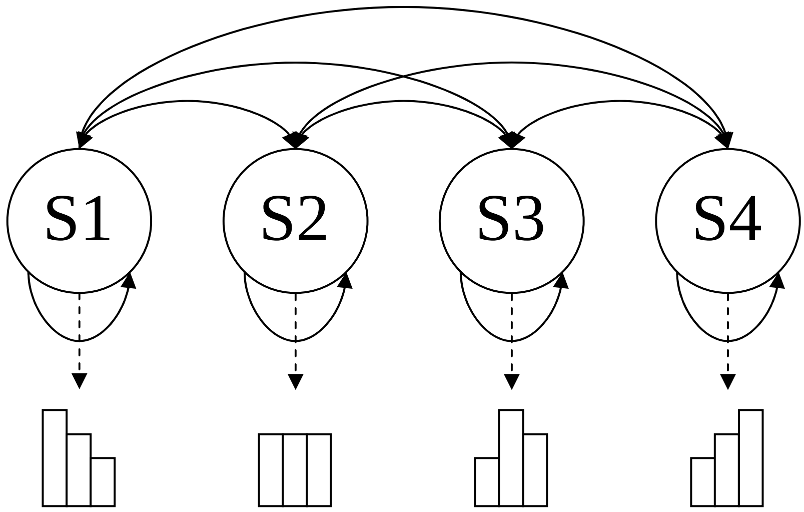

Figure 20. 3-symbol, 4-state Hidden Markov Model

HMMs schemes implemented for this activity model the distribution of error and error-free burst lengths (defined as the number of consecutive bad and well received packets, respectively). Therefore, the length of the intervals is taken as the observed parameter, that is, the symbols that each state can emit according to certain probability distribution.

Since there is not a direct match among states and observable system parameters, the number of states, *S*, can have a significant influence on the accuracy of the model. For that reason it has to be considered as a tuning parameter for specific scenarios to be modelled.

Hidden Markov Models characterize one-dimensional vectors. Thus we have used two different HMMs, one for error bursts occurrence and the other for error-free bursts occurrence.

For the training of the Hidden Markov Model parameters Baum-Welch algorithm [14] has been used.

5.3 Software tools and data gathering environment

The developed network modelling and trace generation toolkit has been integrally implemented in MATLAB. This toolkit also contains statistic, visualization, and other auxiliary functions for the evaluation of the accuracy of the models.

Real RTP level packet loss traces are taken from a single laptop and for a fixed set of transmission conditions (source rate, channel speed, SNR, and RTP packet length). Specific values have been selected according to the network/services scenarios proposed in deliverable D1.1 “User Terminal Requirements”.

Traces are captured by a wireless network analyser (Wireshark/Ethereal [16]). Then, they are formatted with AWK scripts as a two column matrix where the first column contains the length of the error-free bursts, and the second one the length of the error bursts. Typical length of the traces is one million packets (up to 2.5 hours transmission), considered to be long enough to reflect the channel behaviour.

Each one of these traces is taken as input for the training of the three proposed models. Once they have been trained, obtained models are able to generate simulated RTP packet level transmission traces of an arbitrary length, which is one of the main goals for this activity.

Keeping modularity in the toolkit software has been taken as a must. Model training and trace generation interfaces are prompt, easy to use. The main functions are described in the following table:

Table 9 Main functions of the packet-level modeling toolkit and packet-level simulator.

Function name	Parameters	Description
`GModel`	• `Input sequence(s)`	`Trains a simplified Gilbert Model with input data.`
`GModelGen`	• `Gilbert model` • `Trace length (L)`	`Generates an L packets trace according to the model.`
`HMMModel`	• `Input sequence(s)` • `Number of states (S)` • `Number of symbols (N)`	`Trains a S-state hidden Markov model, modelling burst lengths up to N.`
`HMMModelGen`	• `HMM Model` • `Trace length (L)`	`Generates an L packets trace according to the model.`
`ClassModel`	• `Input sequence(s)` • `Cluster length (W)` • `Number of PER classes (N)`	`Trains a hybrid N PER-classes model, with input data divided in W packets length clusters.`
`ClassModelGen`	• `PER classes model` • `Trace length (L)`	`Generates an L packet trace according to the model.`

5.4 Performance and Results

The accuracy of these models has been tested in different scenarios. For each test scenario, a complete set of channel speeds and source rates are considered. Table 10 summarizes the conditions of the tests carried out.

Table 10 Test scenarios

Test Scenario	SNR	IP Packet size	Source rate (Mbps)	Channel speed (Mbps)
Test Scenario #1 *Maximum SNR level 1500 byte long packets*	*Maximum (≈60 dB)*	*1500 bytes*	*3*	*6*
				11
				24
				54
			6.25	*11*
				24
				54
			16	*24*
				54
Test Scenario #2 *Maximum SNR level 512 byte long packets*	*Maximum (≈60 dB)*	*512 bytes*	*3*	*11*
				54
			6.25	*11*
			16	*24*
				54
Test Scenario #3 *Medium SNR level 1500 byte long packets*	*Medium (25-35 dB)*	*1500 bytes*	*3*	*11*
			6.25	*11*
			16	*54*

To measure the accuracy of the different models, we have compared traces generated by the models with the actual traces. Particularly, the average length of the burst of the generated traces is computed and compared with the actual value. In addition, a Chi-square comparison between burst length distributions is performed both for well and bad received packets cases. Chi-square statistic [17] evaluates “goodness of fit” (similarity between measured and modelled values) in terms of frequency (number of occurrences):

$$\chi^2 = \sum \frac{(O-E)^2}{E}$$

where *O* is the observed and *E* the expected frequency. Since only burst lengths with more than five occurrences must be taken into account [18], very long (and rare) bursts are not considered in the comparison. This statistic is basically a sum of relative errors, so the lower the value the better the model fits to real data distribution.

5.4.1 Model parameters selection

As described in section 5.2, the Class Partitioned Markov Chain and the Hidden Markov approaches define a set of configurable parameters (*N*, number of classes, and *L*, length of the analysis cluster, *S*, number of states) that may influence the accuracy of the models.

The value of *N* and *S* indicates the number of states used in each approach. Their values have been determined in order to obtain a trade-off between guarantying a sufficient training (having an excessive number of states could lead to a deficient training) and flexibility (a low number of states can prevent the model from capturing the actual behaviour).

Regarding *L* for the Class Partitioned Markov Chain model, an analysis for a particular transmission conditions is presented. Particularly, for 3 Mbps RTP flow transmission over a 6 Mbps channel (Scenario #1). As can be observed in Table 11 and Figure 21, the error estimation of the mean value is reduced as *L* increases. On the other hand, the Chi-square statistic is higher as *L* increases. Note that using small clusters allows a finer representation of the histogram, and thus it provides a better result on the Chi-square statistic. As *L* increases, the resolution of the analysis decreases penalising the Chi-square statistic, but the mean estimate is more accurate since it is less affected by the effect of the clustering process.

Table 11 Influence of cluster size in Class Partitioned Markov Chain model

Cluster size	**Error in mean estimation**		**Chi-square value**	
	Error-free bursts	*Error bursts*	*Error-free bursts*	*Error bursts*
200	50.32%	48.55%	1711.3	3960.0
500	30.97%	35.60%	2110.8	2411.2
1000	24.02%	22.38%	2655.8	2696.0
2000	14.00%	23.31%	3421.2	2833.9
5000	8.73%	10.59%	5728.4	4756.1
10000	0.64%	11.62%	6975.9	12996.0

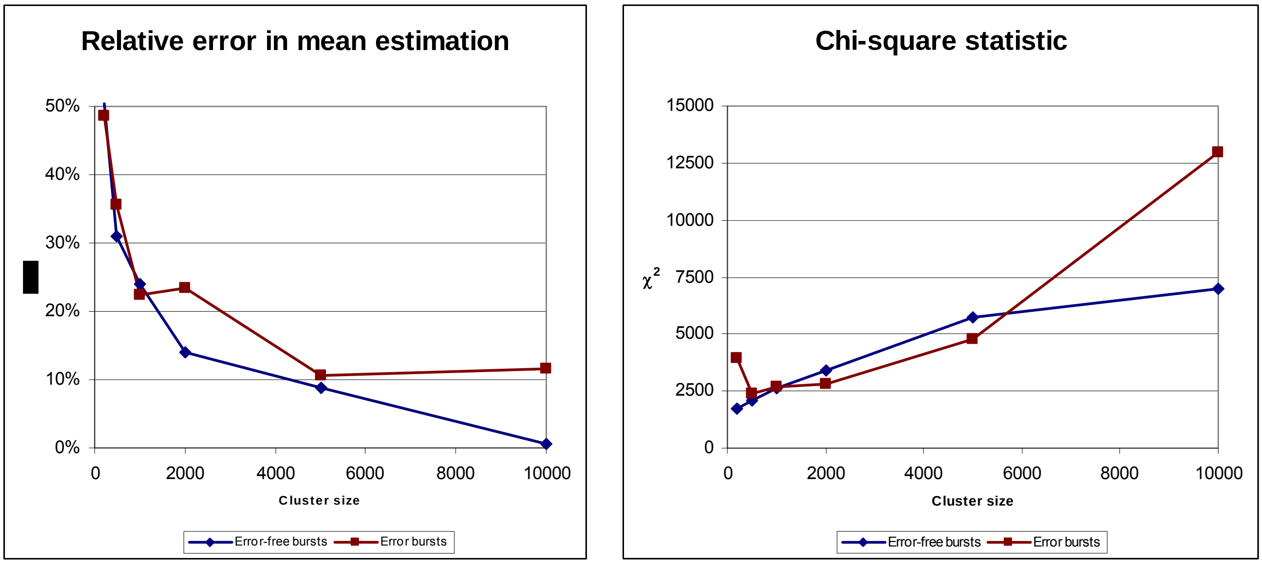

Figure 21. Influence of cluster size in Class Partitioned Markov Chain model

For the rest of experiments, a cluster size of 5000 packets has been selected. This value has been selected as a compromise between the two effects described before.

5.4.2 Test Scenario 1

The first scenario is set to evaluate the proposed models in optimum SNR conditions (around 60 dB). RTP payload is of 1472 bytes, thus obtaining 1500-byte long IP packets (the maximum length without packet fragmentation). Traces involved are around one million packets long (up to 2.5 hours long captures).

Figure 22 shows a typical probability distribution of error-free burst lengths for the transmission of a 3 Mbps RTP flow through a 6 Mbps 802.11 channel. Most of the error bursts are only one packet long, while error-free bursts can have thousands of packets (although in Figure 22 the represented burst lengths have been limited to 500). Average burst length for this trace of around 200 packets.

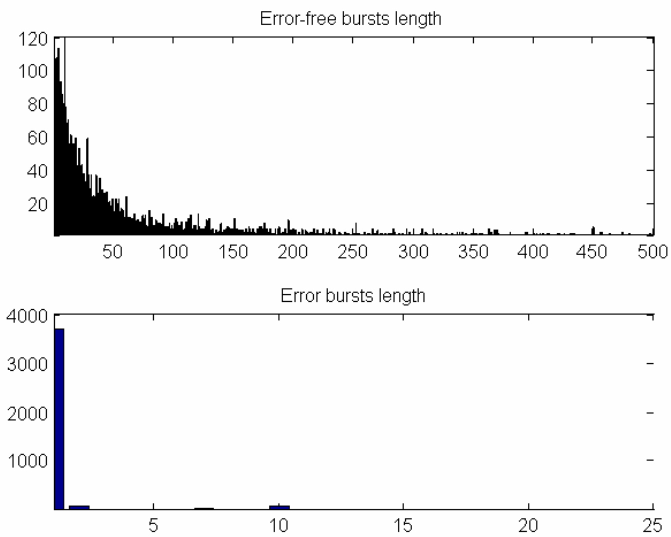

Figure 22. Burst length distributions of the actual trace for 3 Mbps source transmission through 6 Mbps channel.

Figure 23, Figure 24, and Figure 25 show the burst length distributions of a single trace generated by each one of the proposed models.

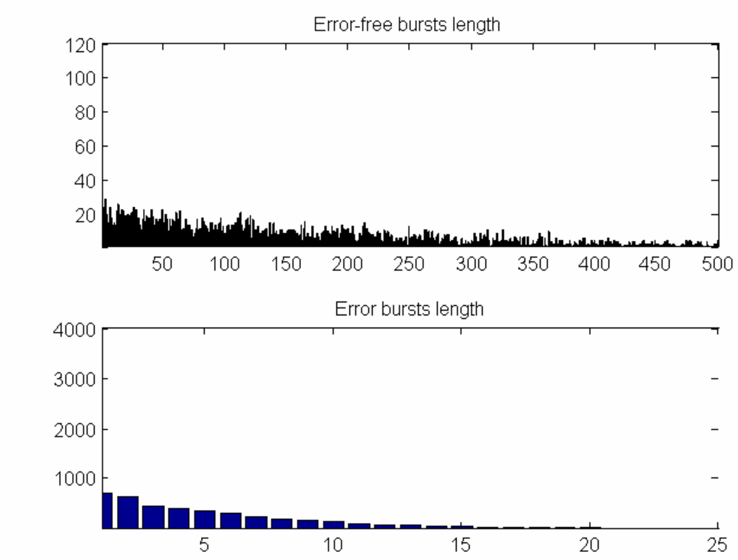

Figure 23. Burst length distributions for trace generated by Gilbert model 3 Mbps source transmission through 6 Mbps channel.

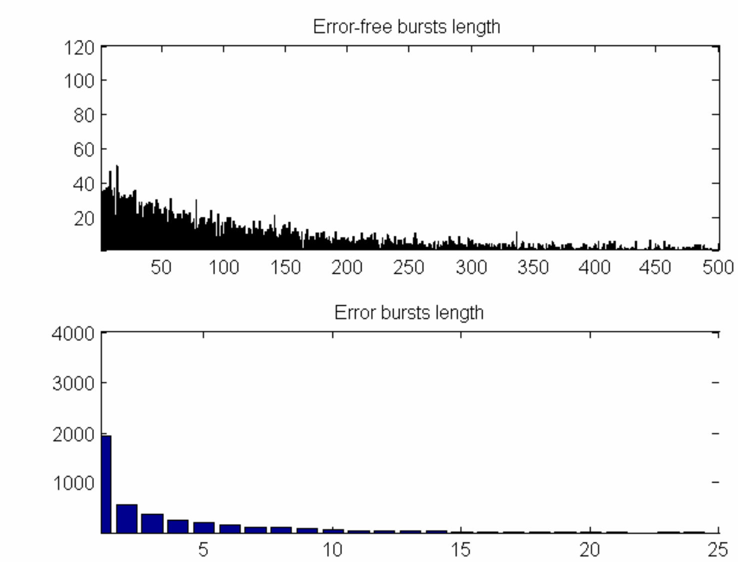

Figure 24. Burst length distributions for trace generated by Class Partitioned Markov Chain model for 3 Mbps source transmission through 6 Mbps channel.

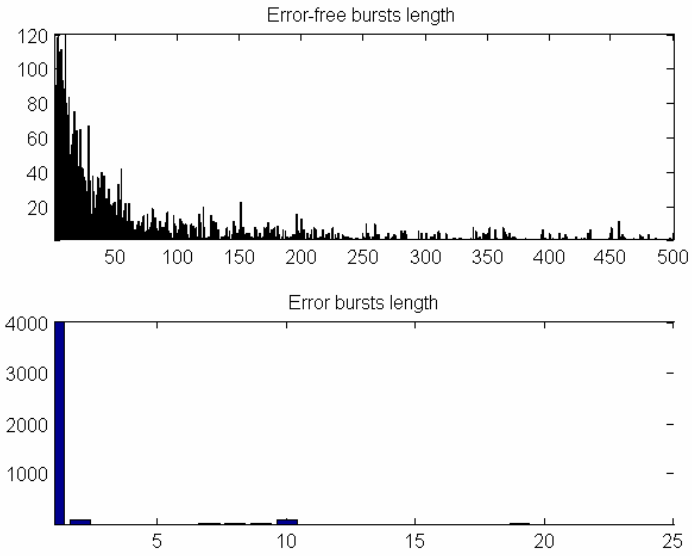

Figure 25. Burst length distributions for trace generated by Hidden Markov Model for 3 Mbps source transmission through 6 Mbps channel.

Table 12 contains the relative error of the estimated mean value of the burst lengths (generated traces) with respect to the actual mean burst lengths (measured traces) for the three model approaches. Table 13 contains the Chi-square statistics obtained for the three models.

Table 12 Relative error of the estimated mean burst lengths for Test Scenario #1

Scenario #1		**Error-free bursts**			**Error bursts**		
Source rate	**Channel speed**	**Gilbert**	**5 PER Classes**	**3-state HMM**	**Gilbert**	**5 PER Classes**	**3-state HMM**
3 Mbps	*6 Mbps*	0.60%	4.51%	10.84%	0.50%	3.31%	14.73%
	11 Mbps	4.63%	48.24%	58.68%	1.08%	56.93%	10.67%
	24 Mbps	1.34%	1.86%	1.51%	6.10%	1.47%	0.03%
	54 Mbps	0.08%	1.54%	3.41%	0.40%	2.23%	4.08%
6.25 Mbps	*11 Mbps*	3.31%	23.36%	23.74%	3.25%	23.57%	34.08%
	24 Mbps	0.72%	12.29%	38.70%	1.96%	4.28%	19.42%
	54 Mbps	1.22%	3.87%	5.29%	2.01%	20.16%	11.62%
11 Mbps	*24 Mbps*	1.29%	1.73%	0.22%	1.95%	3.24%	11.56%
	54 Mbps	1.49%	10.11%	25.07%	1.25%	28.21%	37.10%
Average value		1.63%	11.94%	18.61%	2.06%	15.93%	15.92%

Table 13 Chi-square statistics for Test Scenario #1

Scenario #1		**Error-free bursts**			**Error bursts**		
Source rate	**Channel speed**	**Gilbert**	**5 PER Classes**	**3-state HMM**	**Gilbert**	**5 PER Classes**	**3-state HMM**
3 Mbps	*6 Mbps*	7887.0	5471.0	748.5	21905.0	6769.8	115.6
	11 Mbps	959.4	480.9	64.9	605.6	342.7	85.8
	24 Mbps	7499.3	4119.7	701.1	4483.8	696.6	144.8
	54 Mbps	30835.0	12295.0	793.4	5940.1	3275.2	53.0
6.25 Mbps	*11 Mbps*	28771.0	5375.6	6347.9	3482.7	468.7	295.1
	24 Mbps	5233.6	2784.9	1105.7	5931.4	475.9	52.8
	54 Mbps	5024.2	3260.2	1158.9	6162.7	393.2	44.0
16 Mbps	*24 Mbps*	5213.1	4667.5	829.4	7557.7	6864.6	97.9
	54 Mbps	8758.0	4379.4	976.0	7955.7	517.6	94.0
Average value		11131.2	4759.4	1414.0	7113.9	2200.5	109.2

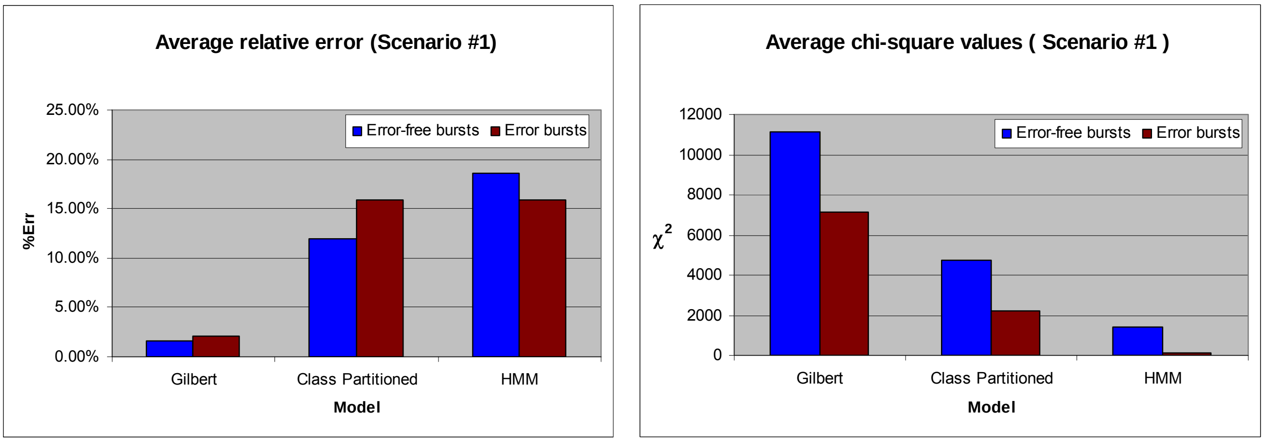

Figure 26. Model performance comparison for Test Scenario #1

5.4.3 Test Scenario 2

In this scenario, the transmission parameters are the same as in the test scenario 1 except for the packet length at IP level which has been fixed to 512 bytes. The aim of this test is to analyze the performance of the models when transmitting short packets. For this purpose a subset of combinations of channel speeds and source rates has been selected.

As in the previous test scenario, Figure 27 to Figure 30 show the burst length distributions of a single trace, real and modeled ones, for 3 Mbps source transmission through a 54 Mbps channel.

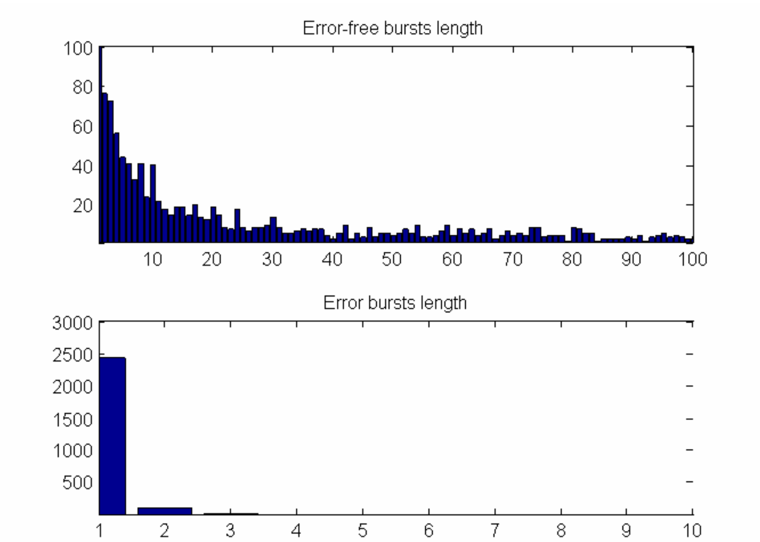

Figure 27. Burst length distributions of the actual trace for 3 Mbps source transmission through 54 Mbps channel.

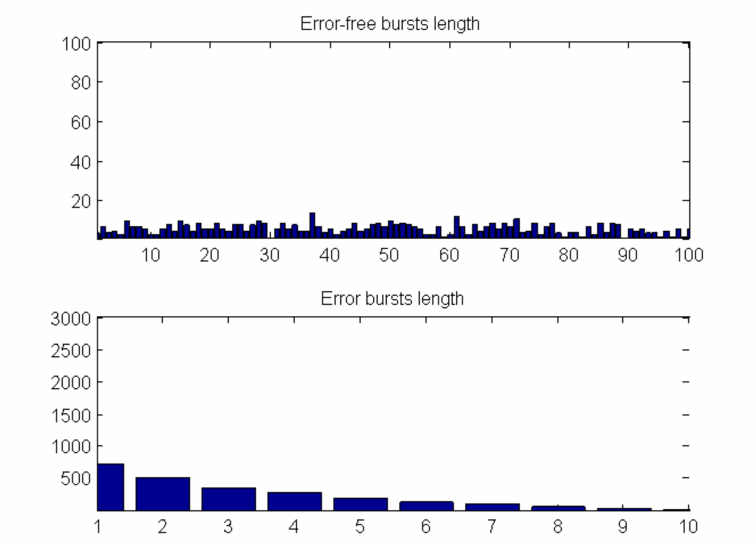

Figure 28. Burst length distributions for trace generated by Gilbert model for 3 Mbps source transmission through 54 Mbps channel.

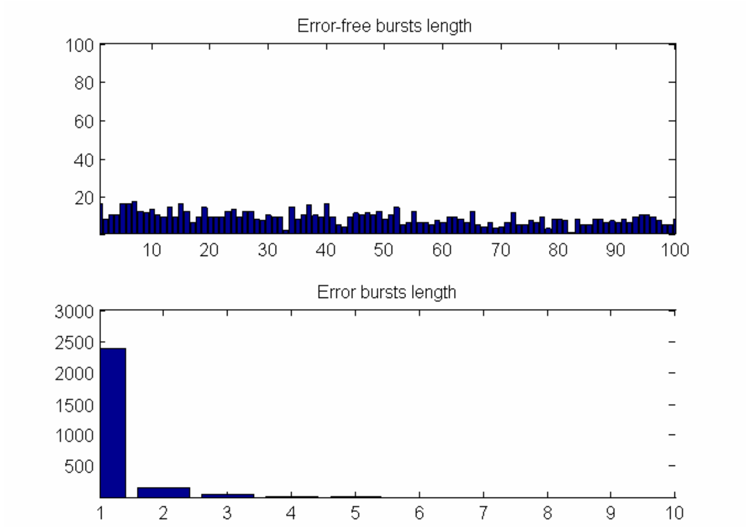

Figure 29. Burst length distributions for trace generated by Class Partitioned Markov Chain model for 3 Mbps source transmission through 54 Mbps channel.

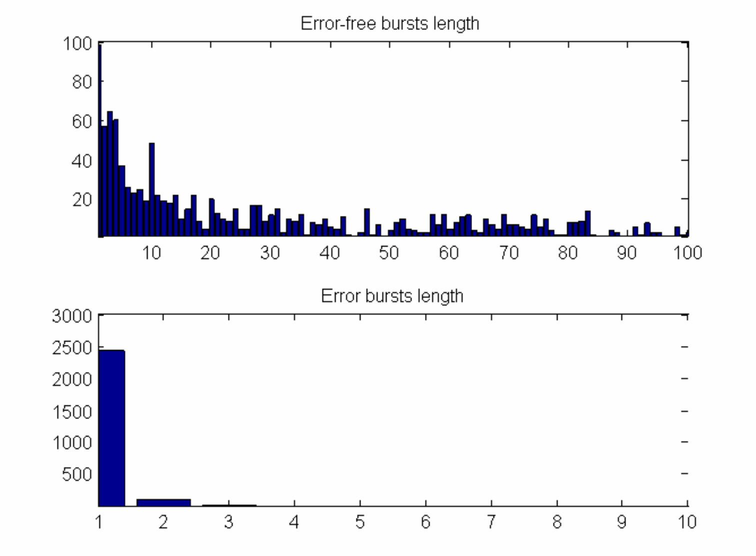

Figure 30. Burst length distributions for trace generated by Hidden Markov Model for 3 Mbps source transmission through 54 Mbps channel.

Table 14 contains the relative error of the estimated mean value of the burst lengths (generated traces) with respect to the actual mean burst lengths (measured traces) for the three model approaches. Table 15 contains the Chi-square statistics obtained for the three models.

Table 14 Relative error of the estimated mean burst lengths for Test Scenario #2

Scenario # 2		**Error-free bursts**			**Error bursts**		
Source rate	**Channel speed**	**Gilbert**	**5 PER Classes**	**3-state HMM**	**Gilbert**	**5 PER Classes**	**3-state HMM**
3 Mbps	*11 Mbps*	0.44%	46.64%	33.79%	7.03%	42.67%	30.06%
	54 Mbps	0.09%	5.76%	10.29%	1.36%	0.12%	11.11%
6.25 Mbps	*11 Mbps*	0.04%	2.86%	0.06%	0.47%	0.07%	0.75%
16 Mbps	*24 Mbps*	0.84%	1.11%	0.27%	0.06%	5.10%	0.71%
	54 Mbps	0.14%	0.08%	0.19%	0.12%	1.87%	1.29%
Average		0.31%	11.29%	8.92%	1.81%	9.97%	8.78%

Table 15 Chi-square statistics for Test Scenario #2

Scenario # 2		**Error-free bursts**			**Error bursts**		
Source rate	**Channel speed**	**Gilbert**	**5 PER Classes**	**3-state HMM**	**Gilbert**	**5 PER Classes**	**3-state HMM**
3 Mbps	*11 Mbps*	1304.5	590.1	200	3145300	169.7	222
	54 Mbps	1002.4	5518.4	73.3	130690	250.6	147
6.25 Mbps	*11 Mbps*	8221.7	5931.7	976	10990	9462.6	175
16 Mbps	*24 Mbps*	6030.4	338.1	39.2	106760	88806.0	159.9
	54 Mbps	15508	15441.0	123.4	2462.3	2150.6	205.9
Average		6413.4	5563.9	282.38	679240.46	20167.9	181.96

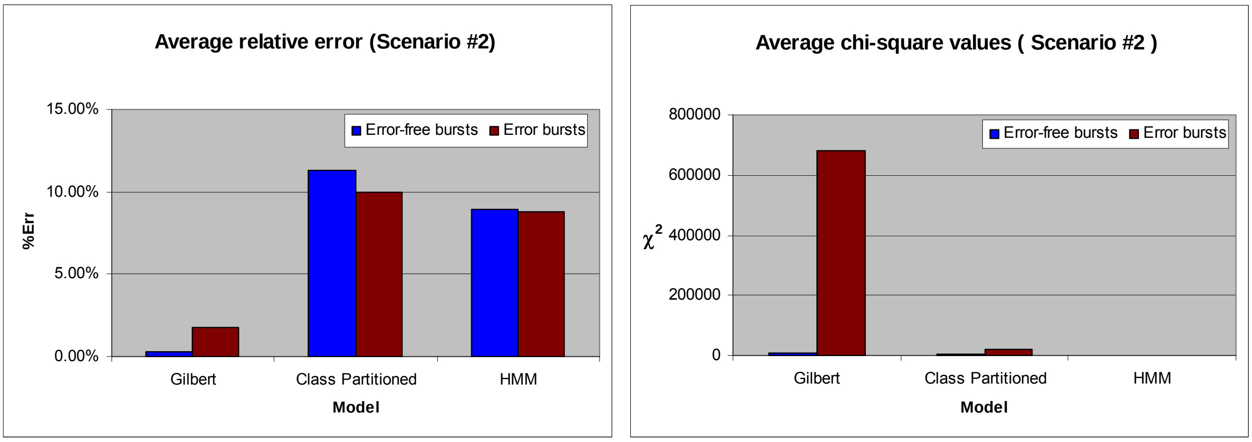

Figure 31. Model performance comparison for Test Scenario #2

5.4.4 Test Scenario 3

In this scenario, the transmission parameters are the same as in the test scenario 1 except for the SNR level. In this case, the SNR level has been set so as to having very heavy packet losses. The purpose of this test is to assess the validity of the models in a lossy environment. With this aim, a subset of combinations of channel speeds and source rates has been selected.

Figure 32 to Figure 35 show the burst length distributions of a single trace, real and modeled ones, for 16 Mbps source transmission through a 54 Mbps channel with a SNR of 35 dB.

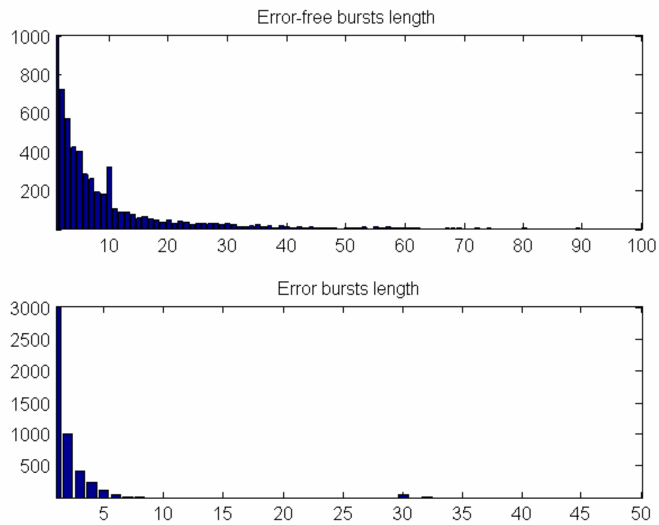

Figure 32. Burst length distributions of the actual trace for 16 Mbps source transmission through 54 Mbps channel (35 dB SNR).

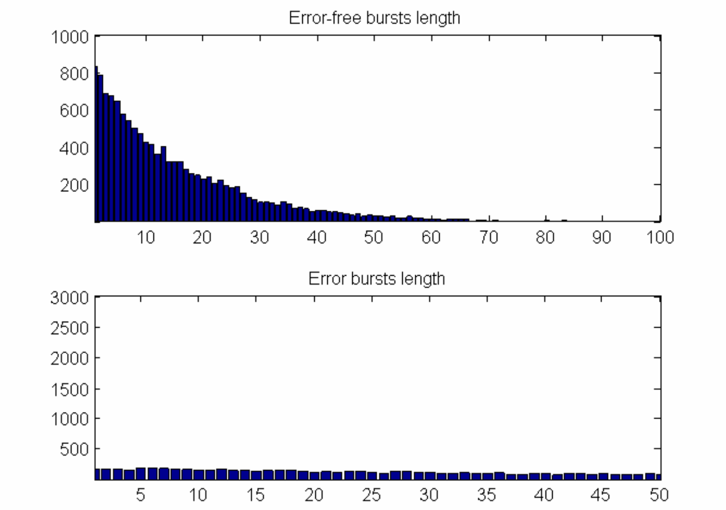

Figure 33. Burst length distributions for trace generated by Gilbert model for 16 Mbps source transmission through 54 Mbps channel (35 dB SNR).

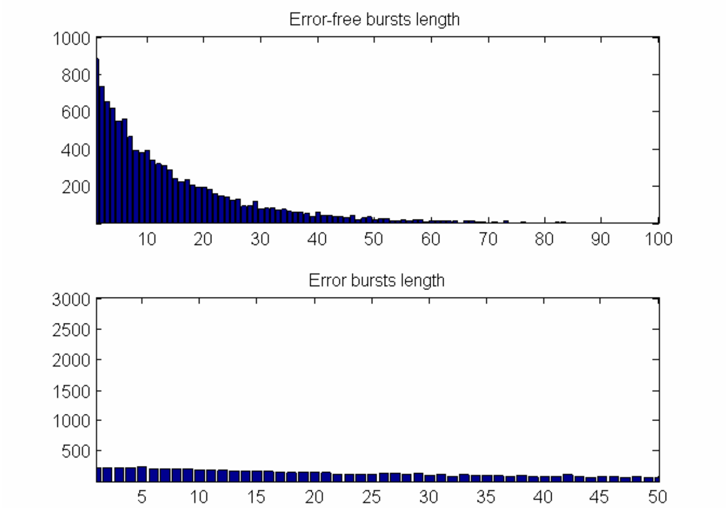

Figure 34. Burst length distributions for trace generated by Class Partitioned Markov Chain model for 16 Mbps source transmission through 54 Mbps channel (35 dB SNR).

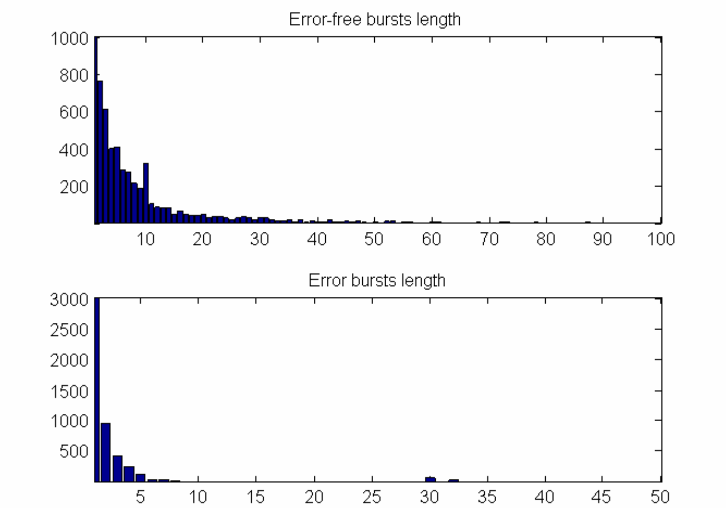

Figure 35. Burst length distributions for trace generated by Hidden Markov Model for 16 Mbps source transmission through 54 Mbps channel (35 dB SNR)..

Table 16 contains the relative error of the estimated mean value of the burst lengths (generated traces) with respect to the actual mean burst lengths (measured traces) for the three model approaches. Table 17 contains the Chi-square statistics obtained for the three models.

Table 16 Relative error of the estimated mean burst lengths for Test Scenario #3

Scenario # 3		**Error-free bursts**			**Error bursts**		
Source rate	**Channel speed**	**Gilbert**	**5 PER Classes**	**3-state HMM**	**Gilbert**	**5 PER Classes**	**3-state HMM**
3 Mbps	*11 Mbps*	0.15%	3.39%	2.29%	0.20%	4.33%	0.03%
6.25 Mbps	*11 Mbps*	0.71%	2.58%	0.36%	0.06%	18.03%	33.06%
16 Mbps	*54 Mbps*	0.90%	0.60%	7.92%	0.64%	0.00%	25.69%
Average		0.59%	2.19%	3.52%	0.30%	7.45%	19.59%

Table 17 Chi-square statistics for Test Scenario #3

Scenario # 3		**Error-free bursts**			**Error bursts**		
Source rate	**Channel speed**	**Gilbert**	**5 PER Classes**	**3-state HMM**	**Gilbert**	**5 PER Classes**	**3-state HMM**
3 Mbps	*11 Mbps*	14896.0	6852.0	86.7	129210.0	65833.0	255.9
6.25 Mbps	*11 Mbps*	1466500.0	1922.1	201.5	4378.1	94048.0	233.1
16 Mbps	*54 Mbps*	160620.0	2646.7	198.7	143420.0	91609.0	283.2
Average		547338.7	3806.9	162.3	92336.0	83830.0	257.4

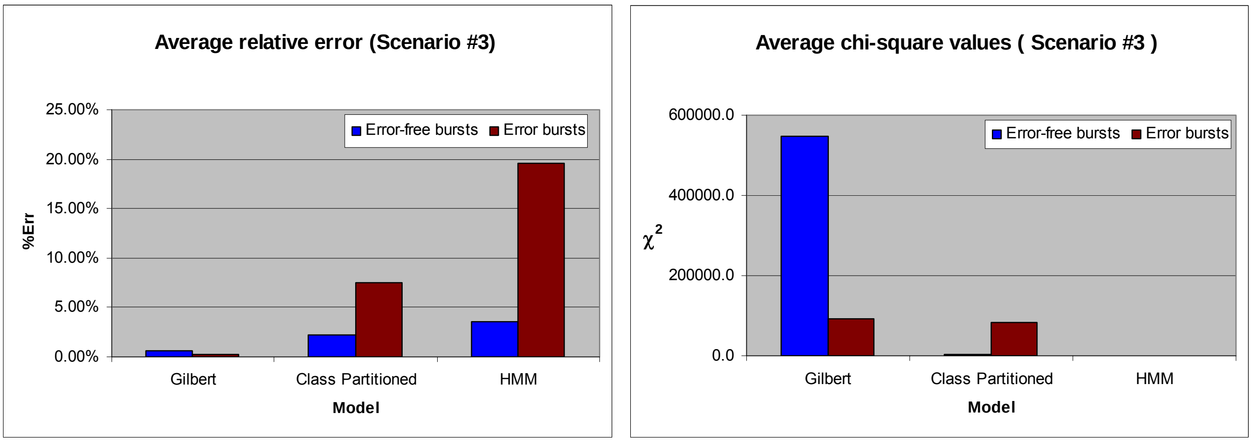

Figure 36. Model performance comparison for Test Scenario #3

5.4.5 Summary of results and conclusions

In most of the tests conducted there is a similar trend in the accuracy of the models. With respect to the estimated mean burst lengths, Gilbert model instantiations provide the best results. This was expected, since its parameters are calculated based on the measured mean values.

On the other hand, HMM instantiations outperforms the other alternatives regarding the Chi-square statistics. In this sense, traces generated by these models will show the histogram closest to that of the actual traces. Regarding the estimated mean burst lengths, the average relative error is about 20 % in the worst case.

The Class Partitioned Markov Chain model shows an intermediate behavior between the other two approaches. The combination of PER-adapted Gilbert models through a Markov chain results in a more flexible model that fits better the probability mass function of the measured traces. However, its Chi-square statistics are worse than those of the HMM approach.

Although results have shown that the Class Partitioned Markov Chain model does not reach HMM's flexibility, the states of the Class Partitioned model can be deterministically determined from the output events. This characteristic may be of interest to develop optimized transmission control strategies.

6 Conclusions

This deliverable describes the WLAN simulation models developed within the project. The first section of this deliverable provides an overview of the WLAN baseband model, its specification and the simulator software validation. In the second section, the simulator software was used to study physical link level performance of WLAN in different HiperLAN/2 channel models and analysis is then given. It can be seen that using WLAN in large area may causes degradation due to channel echoes that are significantly longer than the guard interval of OFDM symbol, as seen in the HiperLAN/2 Channel E. In such cases, care must be taken during the deployment process. The third section gives an overview of the implemented software, encapsulation of IP packets in MAC/PHY protocol layer and the format of the error pattern. The traces generated include all the modulation and coding schemes available in WLAN, with IP packet sizes of 256, 512 and 1024 bytes. Nevertheless, other parameters can be considered easily with the simulator made available to the SUIT partners. The final section describes an alternative discrete-time packet-level channel model developed using real network traces. Analysis of the model accuracy is also presented. The WLAN error pattern and discrete channel models can be used to study optimal video parameters as well as appropriate rate control strategies in other work packages.

7 Acronyms

ATM	Asynchronous Transfer Mode
BER	Bit Error Rate
BPSK	Binary Phase Shift Keying
CRC	Cyclic Redundancy Check
dB	Decibels
DVB	Digital Video Broadcasting
FFT	Fast Fourier Transform
HiperLAN/2	High Performance Radio LAN
HMM	Hidden Markov Model
IFFT	Inverse Fast Fourier Transform
IEEE	Institute of Electrical and Electronics Engineers
IP	Internet Protocol
ISI	Inter Symbol Interference
LOS	Line of Sight
MAC	Medium Access Control
MCS	Modulation and Coding Scheme
MDC	Multiple Description Coding
NLOS	Non- LOS
OFDM	Orthogonal Frequency Division Multiplexing
PER	Packet Error Rate
QAM	Quadrature Amplitude Modulation
QPSK	Quadrature Phase Shift Keying
RTP	Real Time transport Protocol
SNR	Signal to Noise Ratio
SUIT	Scalable, Ultra-fast and Interoperable Interactive Television
TDL	Tapped Delay Line
Wi-Fi	Wireless-Fidelity
WIMAX	Worldwide Interoperability for Microwave Access
WLAN	Wireless LAN
WP	Workpackage

8 References

[1] "All IP Support Requirements," SUIT Deliverable 1.3, document number SUIT_125, 30 September 2006.

[2] "IT++ Signal Processing Library, version 3.10.2," http://itpp.sourceforge.net, May 2006.

[3] "Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications: High-speed Physical Layer in the 5GHz Band," IEEE Std 802.11a-1999, 16 Sep 1999.

[4] "Feasibility Study for Orthogonal Frequency Division Multiplexing (OFDM) for UTRAN Enhancement," 3GPP TR25.892 v6.0.0.

[5] A.Doufexi, S.Armour, M.Butler, A.Nix, D.Bull, J.McGeehan, and P.Karlsson, "A comparison of the HIPERLAN/2 and IEEE 802.11a wireless LAN standards," IEEE Communications Magazine, vol.40, no. 5, pp.172-180, May 2002.

[6] "Fixed and Mobile Channel Models Identifications," WP2.1 SUIT Project Deliverable, July 2006.

[7] S.Armour, A.Doufexi, A.Nix, and D.Bull, "A Study of the Impact of Frequency Selectivity on Link Adaptive Wireless LAN Systems," Proc. of IEEE Vehicular Technology Conference, Vancouver, Canada, pp. 738-742, Sept 2002.

[8] "Part 2: Logical Link Control," ANSI/IEEE Std 802.2, 1998 edition.

[9] "Part 11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) specifications," ANSI/IEEE Std 802.11, 1999 Edition (Reaffirmed 2003).

[10] A.A.Markov, "*Extension of the limit theorems of probability theory to a sum of variables connected in a chain*," Appendix B of R. Howard. "*Dynamic Probabilistic Systems, volume 1: Markov Chains"*. John Wiley and Sons, 1971..

[11] L.R.Rabiner, "*A tutorial on Hidden Markov Models and Selected Applications on Speech Recognition*," IEEE, Vol.77, nº2, pp. 257-285, 1989.

[12] E.N.Gilbert, "*Capacity of a burst-noise channel*," Bell Systems Tech. J., vol. 39, pp. 1253-1265, Sept. 1960.

[13] J.Max, "*Quantizing for minimum distortion*," IRE Trans. Inform. Theory, vol. IT-6, pp. 7-12, 1960.

[14] L.E Baum, "*An inequality and associated maximization technique in statistical estimation for probabilistic functions of a Markov process*," Inequalities, 3:1–8, 1972..

[15] J. A. Hartwell and A. O. Fapojuwo, "*Modeling and characterization of frame loss process in IEEE 802.11 wireless local area networks*," IEEE VTC2004 (Vehicular Technology Conference 2004), pp. 4481- 4485 Vol. 6, 2004.

[16] Ulf U.Lamping, Richard R.Sharpe, and Ed E.Warnicke, "*Wireshark User's Guide: 20002 for Wireshark 0.99.3*," 2006.

[17] J.Neter, W.Wasserman, and G.Whitmore, "*Applied Statistics*," Allyn and Bacon, pp. 482-492, 1992.

[18] W.G.Cochran, "*Some methods of strengthening the common chi-square tests*," Biometrics 10: 417-451, 1954.