An intelligent approach to quality of service for MPEG-4 video transmission in IEEE 802.15.1

by

Guillaume REMY

Submitted in accordance with the requirements for the degree of

Doctor of Philosophy

London Metropolitan University

Faculty of Computing

May 2010

ABSTRACT

Nowadays, wireless connectivity is becoming ubiquitous spreading to companies and in domestic areas. IEEE 802.15.1 commonly known as Bluetooth is high-quality, high-security, high-speed and low-cost radio signal technology. This wireless technology allows a maximum access range of 100 meters yet needs power as low as 1mW. Regrettably, IEEE 802.15.1 has a very limited bandwidth. This limitation can become a real problem if the user wishes to transmit a large amount of data in a very short time. The version 1.2 which is used in this project could only carry a maximum download rate of 724Kbps and an upload rate of 54Kbps in its asynchronous mode. But video needs a very large bandwidth to be transmitted with a sufficient level of quality. Video transmission over IEEE 802.15.1 networks would therefore be difficult to achieve, due to the limited bandwidth. Hence, a solution to transmit digital video with a sufficient quality of picture to arrive at the receiving end is required. A hybrid scheme has been developed in this thesis, comprises of a fuzzy logic set of rules and an artificial neural network algorithms. MPEG-4 video compression has been used in this work to optimise the transmission. This research further utilises an 'added-buffer' to prevent excessive data loss of MPEG-4 video over IEEE 802.15.1transmission and subsequently increase picture quality. The neural-fuzzy scheme regulates the output rate of the added-buffer to ensure that MPEG-4 video stream conforms to the traffic conditions of the IEEE 802.15.1

channel during the transmission period, that is to send more data when the bandwidth is not fully used and keep the data in the buffers if the bandwidth is overused. Computer simulation results confirm that intelligence techniques and added-buffer do improve quality of picture, reduce data loss and communication delay, as compared with conventional MPEG video transmission over IEEE 802.15.1.

Faculty of Computing

London Metropolitan University

To my supervisors

Professor Hassan Kazemian and Professor Phil Picton,

And to my family and all my friends.

ACKNOWLEDGEMENTS

First and foremost, I would like to thank my supervisors Professor Hassan Kazemian and Professor Phil Picton for providing me with the opportunity to complete my PhD thesis at the London Metropolitan University as a French international student.

I especially want to thank my supervisor, Professor Hassan Kazemian, whose support and guidance made my thesis possible. He has been actively interested in my work and has always been available to advise me. I am very grateful for his patience, motivation, enthusiasm, and immense knowledge in intelligence techniques, wireless technology and MPEG video compression. I would like to thank present and past staff members of the Faculty of Computing for giving me an introduction into the Faculty and for sharing their knowledge with me.

I also would like to thank my colleague Doctor Ke Fei You for supporting me during the time we were sharing the same office.

List of figures

List of Tables

CHAPTER 1: Introduction

New technologies have always been an interesting subject. The recent evolution of the multimedia has been a revelation for many people as one discovered the possibility to compress image with a very small degradation of the quality of picture. Since then one has been fascinated by the fast evolution and the quality of the render for even bigger screen and resolution. The evolution of the compression techniques from a global frame point of view to an object oriented compression has been a major evolution in term of efficiency and compression rate.

Communication technique became a very important issue for the last thirty years, firstly with cable and more recently wirelessly. The wireless is really a fascinating subject. The fact that one can receive something from a radio channel is actually amazing. So naturally, interest has been focused on multimedia and wireless. Then learning machine and artificial intelligence became the natural answers to the problem, which is about trying to transmit video over a wireless device. The idea to create an algorithm that can transmit video over a wireless device helped by an artificial intelligence grown to be the best solution to the problem.

This idea was to improve the quality of data reception in a multimedia transmission in real time. The decision has been made to take this idea further by applying Artificial Intelligence to video over a wireless network. As the interest was more focused on short range and low power usage, one decided to stay on the Industrial, Science and Medical band (ISM) and choose IEEE 802.15.1. This standard is based on Bluetooth, and some research has been carried out on multimedia transmitted over Bluetooth. As described later on this chapter, some projects have applied the intelligence technique to carry video or audio files over Bluetooth.

Two of the most interesting points of the IEEE 802.15.1 are its low power consumption and its short range. This standard has been designed to be used in a Private Area Network (PAN). Nowadays, IEEE 802.15.1 is commonly implemented within the "Wi Fi" hardware module. That means IEEE 802.15.1 will become more common. The price should fall and new developments with this communication system will continue. Other short-range area network standards, such as Zigbee, need to be implemented separately to make the device available to communicate with another user.

IEEE 802.15.1 is now installed in all new portable devices such as laptop computers, Personal Digital Assistants (PDA), mobile phones, etc., so it is not necessary to add new modules to use that technology. Furthermore, energy saving is becoming a big issue these days, so using low power consumption devices is very interesting on open-plan offices, PAN and any open-space areas. This wireless connection also allows wire connections to be avoided and is very simple to set up.

In the approach, PAN and multimedia should be working together in order to simplify transmission of video and audio in real time. The problem is that IEEE

802.15.1 has been designed to carry data and voices and not to stream video. The concern was to find a solution to resolve this problem. To resolve it one has to increase the bandwidth, which is not possible as IEEE 802.15.1 has a very limited bandwidth. Alternatively one could try to decrease the amount of data sent during the transmission. MPEG-4 video compression became the compression technique selected. This object-oriented video compression is composed of three frames (I, P, B frames) (developed later in chapter 2). MPEG-4 already has a very good compression rate, but one could reduce the size of the information sending to the sender by dropping the B frame during the peak of transmission. This idea is completed by adding some Artificial Intelligence which can manage the level of the token bucket in the IEEE 802.15.1 hardware. By adding another buffer before the hardware, one could help the token bucket to avoid congestion and soften the data transmission when a problem occurs during communication.

The idea was to use some of this research to build a design by adding a buffer before the token bucket. To regulate the traffic, a Neural Fuzzy Controller (NFC) has been used. If the token bucket is becoming overloaded, the fuzzy controller redirects the streaming flow to the "added buffer" in order to adjust the data rate arriving at the token bucket.

The following represents a good summary of different researches carried out in multimedia transmission; artificial intelligence and wireless communication.

Since the 1940s, when the notion of artificial intelligence was mentioned for the first time, research has been carried out to improve algorithms to learn and analyse data for any system. Communication is becoming a very big issue as people contact each other by electronic means all the time, but reaching someone can be a problem if many people are using the same line to communicate. In 1997, Aboelela and Douligeris proposed a solution based on the implementation of fuzzy rules to increase the utilisation and throughput of the communication network, and to provide a fair distribution of calls, while offering the smallest delay. The results they published proved the interest of implementing this fuzzy system [1].

Another application of fuzzy logic is traffic control in real-time in Asynchronous Transfer Mode (ATM). In 1997, Ascia and colleagues published a paper in *IEEE Transactions on Fuzzy Systems* exploring a new fuzzy logic based system designed to achieve real-time traffic control in high speed networks [2]. The two issues of this paper were; first, the focus on the possibility of hardware implementation in order to avoid network congestion, and second the suitability of applying fuzzy systems to manage a few classes of sources to reach high levels of scalability and cost effectiveness. Their results addressed both of these issues.

Some traditional usage parameter control and policing methods had proved to be inefficient in coping with the requirement of ideal policing. In 1996, the same team had explored an alternative solution based on artificial intelligence techniques. They proposed a policing mechanism based on fuzzy logic that aims at detecting violations of the parameters negotiated. The results showed that the simplicity and the capacity to combine a high degree of responsiveness developed with the fuzzy algorithm were close to the ideal policing management, which is much better than conventional policing mechanisms [3].

Asynchronous Transfer Mode (ATM) networks became very interesting to researchers as the amount of data transmitted increased. Multimedia services over ATM networks need guaranteed quality of service. This guarantee can exist through a few characteristics and services requirements, such as traffic control of call admission and congestion. In 1994, Chang and Cheng proposed another traffic control for admission, a conjunction with congestion based on fuzzy logic rules [4]. Simulation showed an improvement of nearly 10% of the link utilisation while keeping the same quality of service compared with an equivalent non-fuzzy method. In 1996 Chang and Cheng improved their model by managing simultaneously the congestion control and the call admission control on ATM networks. They used two-threshold congestion method to implement the fuzzy traffic controller; thus producing the same capacity admission control method. This improvement used the classic mathematical formulation for the control and mimicked the expert knowledge of traffic control [5].

Douglieris and Develekos published some further research on methods to control congestion over ATM in 1995 and 1997. First, they applied a fuzzy rule-based system to ATM. This method is transparent to sources that are compliant with the traffic contract, effective in shutting down non-compliant sources and quick to respond to mean and peak rate violations [6]. This system showed good results as well as being easy to implement. Next, they improved their previous method by adding an artificial traffic control mechanism using a neural network. The results using the new method showed a real improvement [7].

In a paper published in October 1996 in the *Proceedings of the IEEE* [8], Habib demonstrated how such neurocomputing techniques can be used to address ATM traffic management issues such as traffic characterisation, call admission control, usage parameters control and feedback congestion control. Performance results prove that solutions based on neural networks provide better results, and are simpler and faster than algorithmic approaches. Different types of training techniques can be utilised to "customise" the neural network to a specific application. Predicting the behaviour of nonlinear, time-variant systems can be achieved by back propagation neural networks or radial basis functions.

Another way of implementing artificial intelligence on ATM is to work on the rate control for the available bit rate service class. Hu, Petr and Braun developed Sugeno's method based on a fuzzy Available Bit Rate rate controller [9]. This intelligent approach has been optimised by a traditional gradient search method, in which the parameters of the output fuzzy functions are adapted based on the gradient of a certain control criterion. The simulation showed that low cells loss ratio and high link utilisation are improving.

Liu and Douligeris introduced the idea that the conventional monitoring of the buffer status mechanism to detect congestion in ATM networks is not totally efficient and they proposed an intelligent approach for this method. They designed an explicit congestion notification mechanism for ATM networks using artificial neural networks to estimate the amount by which sources need to reduce their transmission rate. The simulation of their algorithm provides simple and effective traffic control. A reduction of up to 10% of the data loss due to congestion has been proved and the delay introduced by the learning controller is near negligible [10].

In research published by Nordstrom and colleagues in 1995, ATM cell and call levels were upgraded by a neural network-based solution in order to improve connection admission control. This research showed a real improvement on ATM traffic control [11].

Modern communication networks are expected to have hundreds of thousands of nodes to allow connection for millions of users. In such vast telecommunications networks, the amount of data sent and the number of nodes and links are so large that the conventional network control might be not effective due to the high degree of complexity. Park and Lee imagined how a neural network can contribute to provide flexibility, speed, adaptability and intelligent control for new telecommunication infrastructure. The results of their models allowed computers to deal with difficult tasks that could not be performed sufficiently by traditional computer schemes. Their 1995 paper confirmed that neural networks can solve various control problems in high-speed networks [12].

In 1997, Pitsillides, Sekercioglu and Ramamurthy published a paper on a fuzzy explicit rate marking (FERM) traffic flow control algorithm for a class of best effort service, known as Available Bit Rate (ABR). The flow rate is calculated by the fuzzy congestion control module which monitors the average ABR queue length and its rate of change, then uses a set of linguistic rules. Experiments show that FERM exhibits a robust behaviour, even under extreme network loading conditions, and ensures a fair share of the bandwidth for all virtual channels regardless of the number of hops they traverse [13].

More recently, in 2008, a new approach to connection admission control was explored by Olufade, Onifade and Aderounmu [14]. This new design is to keep the network load moderate in order to achieve a performance objective associated with quality of service. They employed fuzzy logic rules in statistical connection admission control by multiplexing of the bandwidth between the peak data rate and the average data rate. The rule defined in this algorithm, is a "min-max" inference method in which the output membership function is given the truth value generated by the premise.

In 1994 and 1995, Tarraf and Habib published research on implantation of neural networks to estimate the probability density function of the traffic [15] [16]. The architecture of the rules was composed of two inter-connected neural networks. The first network learns to obtain the ideal traffic value, while the second is trained to capture the real traffic offered during the process. A comparison of the output of each neural network is made in order to generate an error. This error value is used to shape the traffic back to its original values. The results proved that the algorithm is very efficient in detecting and policing any kind of traffic violation. In 1995 the same authors presented their work on high speed integrated networks using Asynchronous Transfer Mode (ATM) cell switching technique. Quality of service is a requirement for new services such as teleconferencing and television broadcasting. Achieving high utilisation, while keeping the quality of service, is the objective of an ATM traffic management strategy. Neural networks and fuzzy logic can be used to design effective traffic management algorithms by developing adaptive, learning and computational rules. In the paper, the authors demonstrate the interest of using neuro-computing techniques to address ATM traffic management issues such as call admission control, usage parameters control and feedback congestion control. The results were very positive.

In 1998, a group of researchers including the author presented a paper at the IEEE World Congress of Computational Intelligence held in Anchorage, Alaska. In the paper, one could see a way of routing prediction in broadband networks. In this project, a comparison between the "classic" quality of service routing scheme and the intelligent system was developed. The main difference between those two models is the strategic behaviour when paths are selected. The flexibility of the fuzzy approach allows incorporation of enhanced evolution-fuzzy prediction primitives and other algorithms to exploit as much information as possible [17].

Intelligent techniques have also been used for multi-attribute decision making in fuzzy environments. In January 2005, Deng-Feng Li published a paper on a new fuzzy closeness methodology for making decisions in fuzzy areas [18]. The method is based on an aggregating function which attempts to be the closest to the ideal solution. In his paper, Li demonstrated that his method often reached the ideal values. This method of compromise ranking determines a compromise solution, providing a maximum place for the majority and a minimum for the slaves.

Artificial intelligence has been more specifically used on wireless communication to deal with imprecise location based on radio frequency. For example, in 2006, Astrain and colleagues published their research on an indoor positioning approach based on the pattern recognition of the IEEE 802.11 signal strength measurement, using fuzzy logic to deal with vagueness and uncertainty on the trilateralisation based on the signal strength [19]. Tracking and user localisation are considered to provide full intelligent location-based services. Their paper demonstrates that even when the user trilateralisation cannot be as precise as desired, the fuzzy location techniques allow the location ratio to be increased.

Previously, in 2003, Huang, Lai, Tsai, Hsiao and Liu published a paper on a routing protocol proposed for Bluetooth-based mobile ad-hoc networks. The routing tables are maintained in the master devices and the routing zone radius for each table is adjusted dynamically by training a fuzzy inference system. The model showed that the dynamic adjustment of the routing table size in each master device results in much shorter reply time for the routing request, fewer request packets and fewer useless packets compared with two other protocols [20].

In the same year, Homnan and Benjapolakul proposed a new model to adjust soft handoff thresholds dynamically by using a fuzzy inference system [21]. The goals are to decrease the thresholds at low traffic loads in order to give a high quality of traffic channel and to increase the thresholds at the high traffic loads to release the traffic channel in order to support more traffic. In this method, a triangular membership function has been used for the fuzzy rule.

Mobility in wireless communication is of key importance, but it is a big issue as well. Many studies about mobility management have been carried out seeking to guarantee quality of service and to offer advanced services based on the user's location. In 2004, Astrain, Villadangos, Castillo, Garitagoitia and Farina published on a fuzzy method dealing with the problem of determining the propagation path of mobile terminals. As multipath fading and attenuation make it difficult to determine the position of a device, the authors used fuzzy symbols to model the situation, which allowed better working with this adaptive information. A significant improvement of final recognition rate of the path followed by the mobile device was proved in their results [22].

In wireless, ATM and self control are primordial to control the traffic. The work of Al Agha and Labiod sought to improve those two systems in order to increase the resource allocation. In this study, the authors based their approach on the application of distributed artificial intelligence techniques in order to build a self adaptive network within random non-uniform traffic conditions [23]. This project implemented some fuzzy techniques into the wireless ATM system to provide a flexible integration of the multi-agent technique in wireless entities. Simulations presented in the paper prove the interest of implementing intelligent technology in wireless asynchronous transfer mode communication.

Moyle and Watts have studied the possibility of implementing neural fuzzy networks in a personal digital assistant. Their 2002 paper outlined the achievements made in the area of small expert systems. The results showed that a neural network is a very good architecture for a generic problem-solving method [24].

In wireless communications, there is constant striving to improve quality of service. For a summary of several wireless techniques and the proposal for a new framework design for integrating wireless awareness into wireless networks to improve network intelligence for wireless communication networks in future, see the paper by Cheng, presented in 2003 at the 9th Asia-Pacific conference [25].

Connectivity is essential in wireless networks, as one cannot reach the user if the connection cannot be made. In October 2005, Nasereddin, Konak and Bartolacci published a paper on developing a new connectivity decision support system based on connectivity maps generated by a neural network scheme [26]. The proposed approach created a map based on the signal strengths from the active wireless devices. The coverage of the location or the signal strengths were predicted by a trained neural network by analysis of free connection points.

Artificial intelligence has been implemented on wire and wireless connection in many areas, but my research is focused on multimedia communication. Many projects had been developed on artificial intelligence and multimedia. In 1994, Nbousse published a design for a fuzzy controller managing cells generated by voice sources in ATM networks [27]. The conventional control method in the design of an ATM cell rate controller is not really efficient to carry voice, as it is characterised by a high degree of burstiness. The fuzzy scheme used in the paper overcomes this complication by appealing to the ability of fuzzy set theory and logic to handle the complexity. A buffer called a "leaky bucket" has been added in order to improve the performance by using feedback via the quality of service parameters. The fuzzy control rules can improve the performance as shown in Nbousse's results.

In 2000, a paper by Hu and Petr dealt with the design and analysis of an end-to-end feedback flow control algorithm motivated by the available bit rate (ABR) service in ATM networks. The methodology in this project is first to predict the ABR buffer status, then base fuzzy-logic rate control decisions on these predicted values, and finally tune the controller parameters using gradient descent methods. The results showed that the algorithm is stable and efficient and outperforms other proposed ABR system rate controllers. The delay induct is small, so the solution developed here can maintain high link utilisation, provide fair allocations of resources and avoid buffer overflows [28].

Noise and interference are very prejudicial in terms of quality of transmission. Numbers of researchers have tried to deal with noisy areas in voice recognition. In 2002, Heckmann, Berthommier and Kroschel published their research on the integration of acoustic and visual information in noisy conditions yields. The authors were dealing with an adaptive scheme to develop a weighting process for speech recognition in various background noise situations [29].

Fluctuations appear in telecommunications traffic, creating congestion. Most queue management schemes use a fixed threshold or limited number of arrival frames to determine when to allow or discard the entry of frames. The consequence is a vague control and the possibility of not reaching the quality of service required. In October 2006, Lekcharoen, Chaochanchaikul and Jittawiriyanukoon elaborated an alternative model based on fuzzy techniques [30]. A fuzzy control mechanism detected a violation in parameter negotiation. Simulation results showed that on wireless frames, the fuzzy control mechanism improved subsequently high throughputs and dropped frames compared with conventional policing mechanisms.

With a limited bandwidth and increased demand for broadband multimedia services in an advanced wireless network, careful manipulation of handoff calls and new calls becomes very important. In a paper published in 2001 [31], You-Chang Ko and colleagues proposed an adaptive solution using a fuzzy logic controller for distributed call admission control which is an adaptive version of the traditional call admission control. This algorithm increased the performance in terms of channel utilisation, new call blocking probability and handoff dropping probability compared with the conventional algorithm.

In 2006 Palazzi and colleagues introduced their work on a wireless home entertainment centre [32]. This notion is about all the digital entertainment services available in the home, recognising that all these services will have to follow certain common characteristics. This system had to have an integrated wireless connection, it had to be developed and carried through the Internet and finally it had to be in one single media centre. The authors had to develop a new strategy to regulate the concurrent access to the wireless network when parallel applications generate different simultaneous flows. That dynamic solution guaranteed a fast and smooth data delivery for real time streams while maintaining a high throughput for the applications. The results prove that the algorithm significantly improved the connection management in a complex wireless network communication.

The efficient transmission of object-based MPEG-4 video over IP networks with QoS management capabilities was the subject of a paper published in June 2003 by Ahmed, Nafaa and Mehaoua [33]. They proposed an extension of the MPEG-4 system architecture with a new "media QoS classification layer". Their design provides accurate and automatic mapping between MPEG-4 application-level QoS metrics and underlying transport networks such as IP DiffServ. The "media QoS classification layer" uses a neural network classification model that is transparent to application and network layers.

Another control model was presented in June 2003 by Bordetsky, Brown and Christianson [34]. Their model provides response time and bandwidth requirement adaptation in multimedia and application-sharing multipoint IP teleconferences for emerging wireless communications. Their algorithm is based on revealing feedback controls for multimedia call preparation and subsequent real-time connection control. Hierarchical coding techniques permitted them to implement a real-time adaptation at the network layer and above. The proposed adaptive management architecture has been illustrated by a case memory representation of call preparation feedback controls, RTP feedback control tests for providing audio-stream bandwidth adaptation, and configuration of integrated experiments.

Saraireh, Saatchi, Al-Khayatt and Strachan have been working on fuzzy and hybrid genetic-fuzzy approaches to assess and improve quality of service (QoS) in simulated wireless networks by sending three multimedia files in real time [35]. A fuzzy inference system evaluated quantitatively the QoS provided by the networks for each application. Two methods to improve the networks' QoS were developed. One method was based on a fuzzy inference system mechanism and the other used a hybrid genetic-fuzzy system. Both methods determined an optimised value for the minimum contention window in IEEE 802.11 medium access control protocol. Their contention window affects the time period a wireless station waits before it transmits a packet and thus its value influences QoS. The results using the two methods improve the average QoS for multimedia file transmission over wireless networks.

In a paper presented in 2005, Chao-Lieh Chen and Po-Chien Hsiao proposed to perform control over the contention parameters in the IEEE 802.11 media access control layer, to enhance QoS in wireless communications, by developing an effective, efficient and distributed fuzzy control algorithm. The proposed fuzzy control algorithms reserve bandwidths by controlling the delays of mobile nodes in the network. The proposed fuzzy control comes from generalised fuzzy automata theory. Furthermore, their project also proposed a fuzzy mean approximation scheme that catches up delay variations and issues controls at adequate time instances. Therefore, the reference target delay can be quickly tracked, the settling time is enormously decreased, the bandwidth constraint is met, channel utilisation is enhanced, and the quality of the network is assured. The results of this system adapt effectively and efficiently the parameters of the bandwidth specifications [36].

It is well known that if a cellular network service system is shared by users with different characteristics, the overall system performance can be improved by denial of service requests even when the capacity for success exists. In 1997, Yener and Rose defined local call admission policies that make admission decisions based on partial state information [37]. They used a genetic algorithm to reach the best local call admission and compared it with small and large systems. Their results show the threshold policies were increasing by about 50%.

Providing efficient access to a large user population with variable service requirements in wireless communications networks poses a very challenging problem. In June 2002, Bandara, Shen and Nurmohamed proposed a fuzzy resource allocator to facilitate the efficient allocation of network resources in the wireless domain. In order to let the fuzzy allocator optimally allocate the remaining resources to non-real-time traffic, they used effective transmission rate statistics of non-real-time traffic sources as a measure of fading channel conditions. Their simulations showed that the fuzzy allocator can reduce delay and incurs fewer retransmissions for non-real-time traffic [38].

As is well known, transmission of video data over dynamically connected ad-hoc networks is challenging. Kwan, Doançay and Jain sought to deal with it by developing a novel multipath selection scheme using fuzzy logic, depth first search and feedback information in order to determine a set of shortest paths between source and destination. This scheme allows layer coding and multipath description coding (the two most popular source-coding solutions for this task) to achieve a higher performance in video playback. The results show that the optimal balance between effective use of network resources and real-time video requirements has been reached [39].

A self-organising fuzzy controller and a rule-based fuzzy controller have been applied to transmit MPEG multimedia file over ATM networks. In 2006 Kazemian developed a new self-organising fuzzy controller to eliminate excessive delay or loss of the variable bite rate data sources at the reception end. Here, the algorithm adjusts the traffic-shaping buffer output rate to enable the encoded video to follow the leaky bucket's contract prior to entering ATM. A rule-based fuzzy controller is used to police the data rate entering the traffic shaper in order to prevent any transmission problem. The simulation results show that this design reduces delay or data loss at the receiving end compared with a conventional rate-policing mechanism in ATM [40].

The performance of call admission control and traffic policing mechanisms has been explored in a paper by Koutsakis and Paterakis in 2004 [41]. This study was carried out in order to transmit multiple quality encoded videoconference images over a wireless ATM channel of high capacity, depending on the user's needs and requests. Both the CAC algorithm and the traffic-policing mechanism are novel mechanisms proposed for the first time in this paper. For their implementation, the design uses an estimation of the equivalent bandwidth of the videos, which has been introduced in the past. Focused on both MPEG-4 and H.263 coded films, the results of their scheme prove a higher aggregate channel throughput, and a preservation of the very strict QoS requirements of the video traffic.

Congestion management in cellular networks is very important for mobile network operators. They utilise improved techniques for enhanced capacity management, but the number of mobile subscribers is rapidly increasing and additionally, new data technologies for wireless access add extra traffic to the already overloaded networks. In their research published in 2004, Kyriazakos and colleagues presented a resource management system for increasing the efficiency of cellular networks during overloading periods [42]. To evaluate the process, a set of trials was performed focusing on alleviating multimedia congestion problems. Results showed real improvements in the network's capacity availability when the presented system intervenes.

User requirements, device capabilities and network characteristics are very important in mobile intelligent multimedia presentation systems. Those presentation systems, which include multimedia content accessed across Hyper Text Transfer Protocol or Real Time Transfer Protocol, are particularly reliant on the capabilities of the connecting mobile network and in particular on the real-time constraints. Solon, Curran and McKevitt developed a design that gives primary consideration to the wireless bandwidth fluctuation in order to control cross-modality adaptation. Their algorithm used a fuzzy inference system to control cross-modality adaptations between video and audio. Particular focus is given to the fuzzy inputs, fuzzy control rules and fuzzy outputs which have been utilised in decision making. The main difference from other approaches is that their design was focused on controlling the bandwidth to determine cross-modality adaptations in a mobile network environment. A very good response was shown in the paper published in 2006 [43].

Fluctuation of wireless channel conditions can add a significant amount of delay to video packets and cause them to miss their play-out time. Jiang and Kleinrock published an article in 2000 in which they discussed techniques for transmitting smoothed video more efficiently over a wireless network [44]. If a video is smoothed, it is possible to selectively deliver packets that are delayed to reduce the impact of the missing packets on video quality. An effective packet-selection algorithm has been carried out to accomplish the goal. This algorithm determined whether to transmit a packet based on channel conditions as well as the likelihood that higher priority layers in the rest of the video will be delivered on time. The results showed that the performance of the algorithm optimises transmission over a wide variety of channel conditions.

Bluetooth is a type of wireless communication. It is becoming a part of the "Wi-Fi", as it is now implementing in the same module as "Wi-Fi", but the major difference is its low power consumption and the very limited bandwidth. Many research studies have been carried out on the possibility of transmitting multimedia files over Bluetooth with sufficient QoS parameters. In 2002, Khan, Wall and Rashid devised a new hardware and software design created to increase the quality of communication within a Wireless Personal Area Network on the Bluetooth protocol stack as a carrier for real-time multimedia communication. Tests done in the laboratory shown very good results [45].

The automatic repeat request (ARQ) algorithms implemented by default on Bluetooth are not really suitable for multimedia transmission. The scheme creates missed displaying and decoded deadlines. Razavi, Fleury and Ghanbari contributed to improved video distribution over Bluetooth by proposing a new design using fuzzy logic control of the ARQ, based on send buffer fullness and the head-of-line packets deadline [46]. The algorithm presented considers delay constraints of the video stream and at the same time avoids send-buffer overflow. The tests have shown an improvement of at least 4dB in video quality compared with the conventional schemes.

Moving Picture Expert Group is an organisation well known in video compression. MPEG and Bluetooth have been used together to improve QoS in video streaming using a fuzzy approach. Kazemian and Meng published a solution in May 2006, adding a fuzzy control system introduced at the Host Controller Interface (HCI) [47]. The system uses a buffer to prevent excessive back-to-back cells. The output bit rate of this buffer is managed by a fuzzy-ruled controller. Another set of rules managed the input bite rate to optimise the loading of the buffer. The results showed a marked reduction of the data loss. The same authors developed another model based this time on a neural approach in the design. The project included a buffer to prevent overflow. Neural fuzzy rules managed the input and the output bite rate of the buffer to ensure the multimedia stream from the host conforms to the traffic conditions of the Bluetooth channel during the communication time. Results on this simulation significantly reduced excessive delays during transmission [48].

After one has reviewed all these papers on Artificial Intelligence (AI), some ideas came to mind. One of them looked very realistic and possible. At the beginning, a different design through Simulink to simulate the hopping frequency of the IEEE 802.15.1 has been developed. After that, a simulation of a transmission of a simple bit, still using Simulink, and a simulation of a voice file transfer have been done. Modifications to the model have been done to develop the Matlab code in order to build up the idea. Two different models have been completed, one with Rule-Based Fuzzy (RBF) and Neural Network Controller (NNC) and another one using Variable Bite Rate (VBR) transmission. These two models have been developed in order to be able to compare and easily demonstrate the improvement of the design in terms of quality of picture as well as reduction of data loss and excessive delay.

The results, in chapter 5, show the advantage of using learning technique in video transmission over short-range area networks. One could easily demonstrate that the burstiness and the overflow in the token bucket have been considerably reduced by the intelligent system.

This thesis will cover general Information on the technique used to design and improve my algorithm.

Initially, the video compression technique has been developed, here MPEG-4. The MPEG-4 has been selected as much research has been carried out on this video compression technique and it has become the standard for video over IP and digital television. All the relevant studies described above prove that MPEG-4 is very stable, and usefully reduces the size of multimedia files.

Secondly, a general but technical information covering the wireless standard used in this project is offered. The IEEE 802.15.1 was based on the original Bluetooth but standardised by the IEEE group. IEEE 802.15.1 is fully compatible with Bluetooth as it was built on it. IEEE 802.15.1 is, as Bluetooth was, becoming ubiquitous in new technology such as mobile phones, Personal Digital Assistants (PDA), laptops, Private Area Networks (PAN) etc. As shown, a great deal of research has been done on this subject to improve the transmission of the data as well as developing new algorithms to use this technology on different devices.

All the evolution mentioned in this introduction uses Artificial Intelligence. Artificial Intelligence has been applied in many areas to improve reception, to increase quality in data transmission and many other aspects. Artificial intelligence has the capacity to learn from events in order to improve subsequent results.

One can prove on all these researches on Artificial Intelligence the vast possibilities of this technology. After reading many papers on Artificial Intelligence applied on wireless or Bluetooth, and on multimedia, the idea of applying Artificial Intelligence to MPEG-4 over IEEE 802.15.1 in order to improve quality of picture came up. This algorithm will be explained in detail in chapter 4, and results of the simulations will be explained in chapter 5. The results clearly show reduction of the overflow during transmission.

CHAPTER 2:

MPEG-4 Video in IEEE 802.15.1

2.1. Concept of video compression

A complete modification of the media industry has occurred in the last fifteen years. New techniques have been created to improve audiovisual contents and the research in audio and video material has nowadays taken a fundamental position in technology. Video compression especially is becoming very important for researchers and engineers. Video compression makes possible to use, transmit, or manipulate videos easier and faster.

The main goal in video compression is to minimise the weight of the files and maximise the quality of the reconstruction. An obvious way to reduce the amount of data is to avoid unnecessary data. In order to do this, one could use and develop two main features: The suppression of the redundancy of data or to deal with the limitations of the human visual system.

A video can be treated as a temporal, spatial and frequency side, which means, the redundancy can be treated from these three angles. If you are talking about a frame, some pixels are similar or very close to the next one. This similarity can be interesting to look at. But moreover, if you are looking at two or more frames in a video sequence, one could see that loads of information is similar from one to another frame. This redundancy can be used to compress the video sequence without losing information.

We all know that the human body is a genius machine, but it has some limits and one could use these limits to reduce the information coded if it is not suitable. For example, the human vision cannot completely cope with more than 25 frames per second, there is no need to transmit more. One can as well limit the number of lines on a television screen for example or one can limit some colours correction within the compression as the eyes are not able to see it. So in order to optimise the compression the limit of the human eyes and brain is really useful as one can remove irrelevant information.

New compression techniques have been created in the past few years. But they more or less all follow the same algorithm and process. This general design is shown on the Figure 2.1. This figure represents the block diagram of a common encoder.

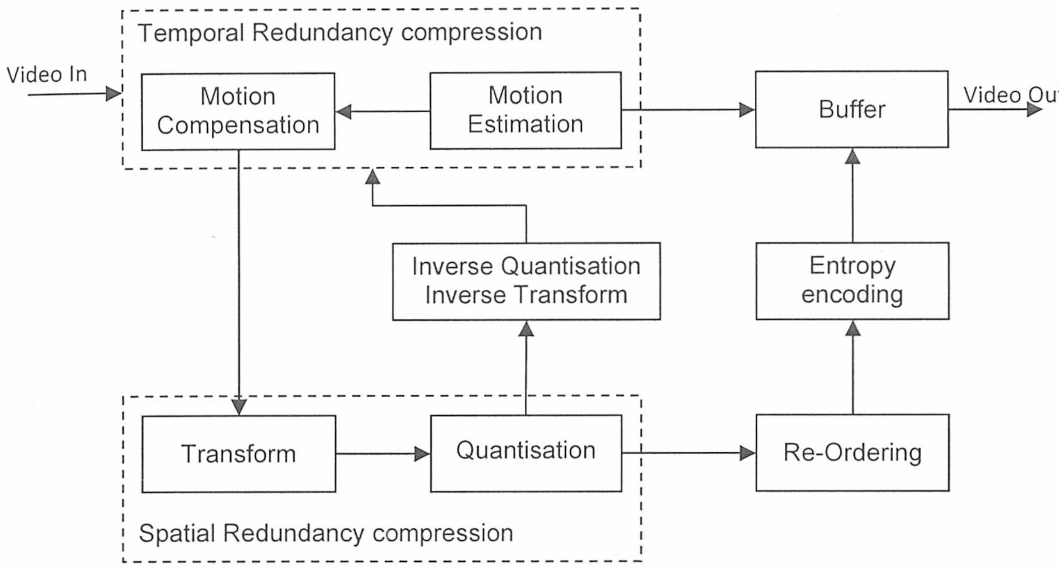

Figure 2.1: Block diagram of a common encoder

In this section, one will firstly discuss the temporal redundancy, then one will discuss motion estimation and motion compensation. After a sub section will be devoted to the spatial redundancy, in that section one will talk about quantisation, discrete cosine transform, re-ordering and discrete wavelet transform. The discrete wavelet transform will be then developed deeply in the followed section. Finally the entropy coding will be mentioned and explained in section 2.1.5.

2.1.1 Principle of Temporal Redundancy Compression

If one observes a video sequence, composed as one knows of consecutive frames, one can easily find that the content of a frame does not vary much from the previous one. These changes are mainly based on the luminance of the picture and some details of the content of the frame.

Then a reduction of the redundancy is possible to light up the data. Inter-frame prediction coding use the technique of defining a reference frame to encode the next upcoming pictures in order to avoid the coding of unchanged parts. This process can be explained in two different steps: firstly one gets the reference picture and based the predicted frame on this referenced picture. The second step is to analyse the difference between theses two pictures. In this analysis, one transmitted the parameters used during the prediction; motion estimation is a part of the prediction process. After motion estimation, the action of computing the residual is called the motion compensation. Motion estimation and compensation is developed in section 2.1.2.

At the encoder, a loop gets each frame back and precedes it as a reference for the following coded frames. 'Transform quantisation' step is doing the same work but inversed. If the frame is encoded on its own, one call this frame 'Intra-frame', in this case the time is not considered in the process. If no temporal redundancy is used, one has a simple static image compression that generates a simple succession of image. If one introduces a temporal factor one obtain an 'inter-frame' as one generated the frame through other frames.

2.1.2 Motion estimation and motion compensation

On a video sequence there is a huge probability to have a high level of redundancy between two consecutive frames. Except if one has a new and completely different action and background, the differences between a frame and the next one are very low. The principle of the temporal compression is to fully encode the first frame as a reference and only encode the difference between the following pictures and the reference frame. If the value is 0, the process will understand this value as a fully similar image as the reference frame and then not encode it. The value will be significant for moving objects, zoom, and luminance changed. In order to reduce even more the amount of data coded, one can estimate the motion between two frames. This will create a motion compensation of the picture and then the differences between the two frames will be the changes between the compensated image and not the original uncompressed image. The better motion estimation is made the better accuracy one can obtain to the motion compensational frame.

2.1.2.1 Motion estimation

Motion estimation has been studied many times especially in video coding techniques [49]. The goal of the motion estimation is to match as good as possible the characteristics of the block of the current picture to the block of the reference picture. Several methods of motion estimation have been developed. On the following part, one will discuss about the most common methods of block matching. The following parts explain some of those methods.

2.1.2.2 Cost function

One of the criteria for group estimation methods is the cost function [49]. The cost-function depends on the parameters decided to evaluate the macroblocks and describes the degree of similarity between two blocks.

In the following equations, the pixels of the current macroblocks are indicated C(x+k, y+l) and the pixels in the reference picture are R(x+i+k, y+j+l).

Mean Absolute Error MAE(i,j) is minimised where

$$MAE(i,j)=\frac{1}{MN}\sum_{k=0}^{M-1}\sum_{l=0}^{N-1}\left|C(x+k,y+l)-R(x+i+k,y+j+l)\right| \tag{2.1}$$

The coordinates (i,j) for which mean absolute error is minimised define the motion vector.

The Mean Squared Error (MSE) can also be chosen as a cost function:

$$MSE(i,j)=\frac{1}{MN}\sum_{k=0}^{M-1}\sum_{l=0}^{N-1}(C(x+k,y+l)-R(x+i+k,y+j+l))^2 \tag{2.2}$$

The correlation between blocks is another way of comparing matching ratio.

Pixel Difference Classification is another way, by counting the number of pixel similar between two blocks. The objective of this method is to obtain the bigger value.

$$PDC(i,j) = \sum_k \sum_l T_{i,j}(k,l) \tag{2.3}$$

Where given a threshold:

$$T_{i,j} = 1 \text{ If } C(x+k, y+l) - R(x+i+k, y+j+l) \leq t \tag{2.4}$$

otherwise $T_{i,j} = 0$

Another possibility is to evaluate the quantised matching pixels instead of the matching pixels themselves. This technique is called the Difference Pixel Count criterion (DPC)

A good solution would be to combine different criteria [49]: for example a first set of motion vector can be defined by a simple matching criterion, and then one could refine the results by using a more accurate criterion. The search method could be one of the more accurate methods that one could use the refine the result.

• **Example of some block matching algorithm**

Many studies have been carried out to evaluate the different block matching algorithm, for example in 2004 [50]. One could find in that study that for an optimum result, the best matching macroblocks should be performed on the whole picture, which is very difficult to do. So I decided to describe a search region around the actual location of the macroblocks (M by N) in the current frame, in order to do the block matching more efficiently.

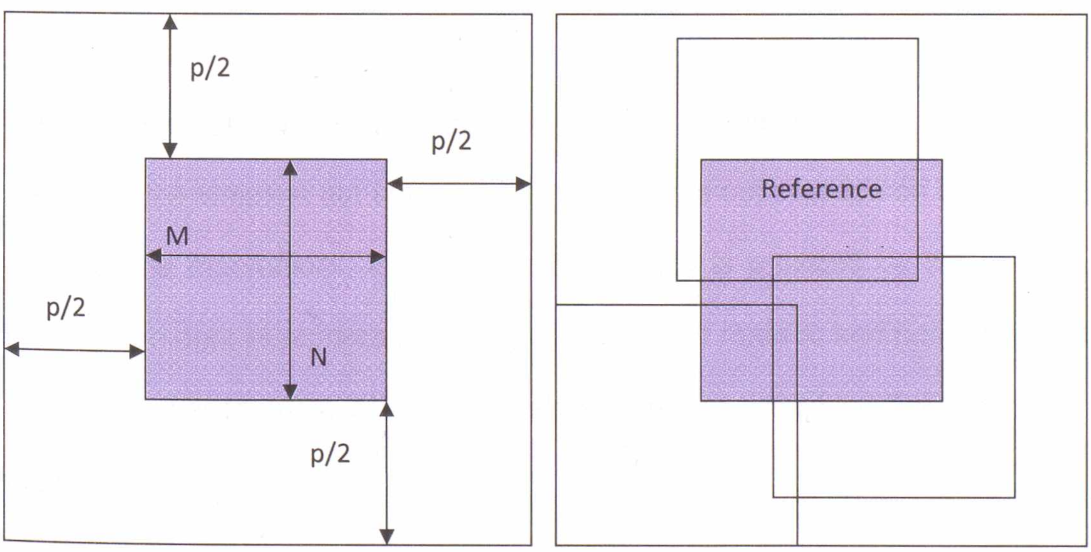

Figure 2.2: Search region and macroblocks to analyse the matching with the reference macroblock

As I have shown, the cost function algorithm produces the best match, by evaluating all locations which need lots of processing resources. In order to reduce this huge demand, a faster algorithm or a reduction of the number of search location should apply. On the following part I will describe some of the well known sub-optimal block matching algorithms. I will first describe the full search algorithm, the three steps search, the Parallel Hierarchical one dimension search, the two dimension search, the four steps search and then the very effective diamond search.

2.1.2.3 Full search algorithm

This algorithm does the block matching on the whole reference frame. This method is the simplest but unfortunately there is not optimisation as there is not edge for the macroblock. Even the best accuracy is provided by this system, the huge amount of process makes this solution not realistic and not often used as it is. However, most of the other algorithms are based on the full search algorithm.

2.1.2.4 Three-step Search

The three-step search (3SS) has been made to simplify the full search algorithm. The result is a very quick and very simpler algorithm but the image quality is visually decreased. In this algorithm, I am limited at three the number of reach to get the best block matching. I am looking at the most similar block from the frame used as a reference. In Figure 2.3, one has a representation of the 3 steps.

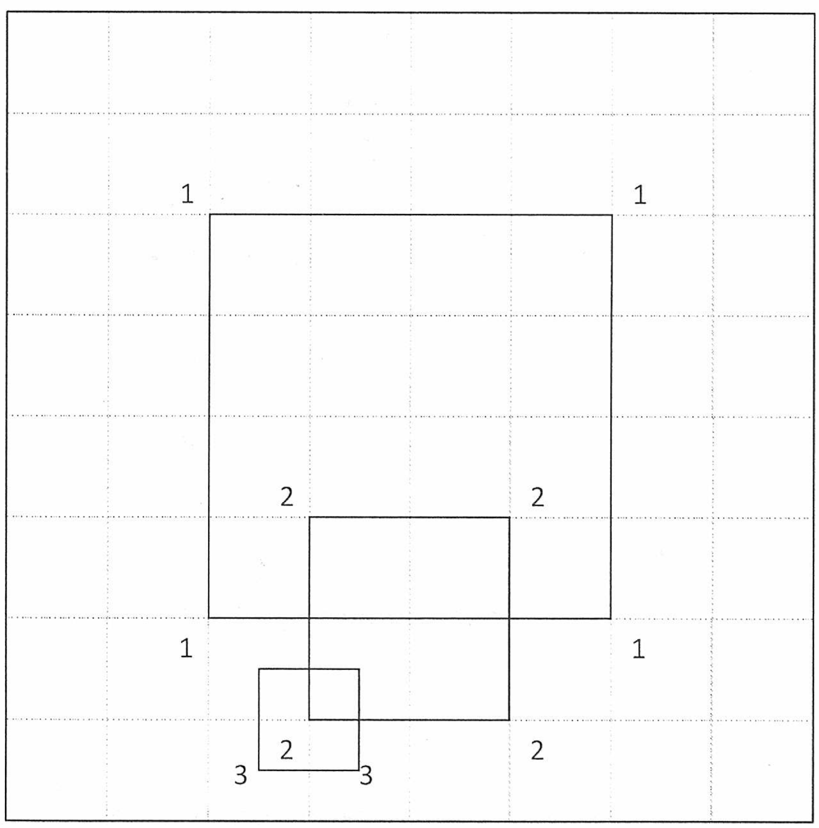

Figure 2.3: 3 Steps Search algorithm.

This method is based on three steps (excluded the initialisation): First, I initialise the motion vector in the centre. Then I analyse around the centre to get the minimum. Second step of the analysis, I get the minimum as the new centre of a new analysis. Then finally I get the new minimum to get the final minimum.

2.1.2.5 Parallel Hierarchical One-Dimensional Search (PHODS)

The specification of this method is to do at the same time and independently the search along the two dimensions of the frame. This method is a very efficient technique in term of velocity and time efficiency, but it gives bad results in term of quality of picture. As this algorithm is working in parallel, the regularity of the flow of data is noticeable [49]. This system is quicker than the 3SS but as the quality of picture is worse, it is not used very often.

2.1.2.6 Two-dimensional Logarithmic Search

The Two-dimensional logarithmic search is another way of searching for block matching. In that system I am decreasing the number of iteration and I am searching at specific location only.

I start from the middle of the block to match, then I am looking for the matching criteria in 5 different locations (centre, north, south, east, west) to a distance D from the centre. Then the minimum matching criteria is used as the centre to the next analysis, for this next research I are reducing the distance D at log(2)D when the minimum value is at the centre of the pattern or if the centre of the pattern is at the edge of the frame.

2.1.2.7 Four-Step Search (4SS)

This algorithm is based on the 3 steps search method, but here the cost function is computed with nine checking points on a 5 x 5 window in the first step instead of a 9 x 9 window in the 3SS. If the centre is the best match the process stops, otherwise the centre is moved and the same process is repeated. When the best match is found at the centre, search window is reduced to 3 x 3 and the search stops.

2.1.2.8 Diamond Search (DS)

The diamond search try to find the minimum value for the matching criteria by analysing 9 points shaped as a diamond. As soon as the minimum value has been found, this point is used as the centre for the next analysis. And then one continues until the minimum value is not at the centre. Thereafter one does a final analysis with a smaller diamond shape to determine the estimated minimum for the frame. This method is very accurate and useful to find a global minimum.

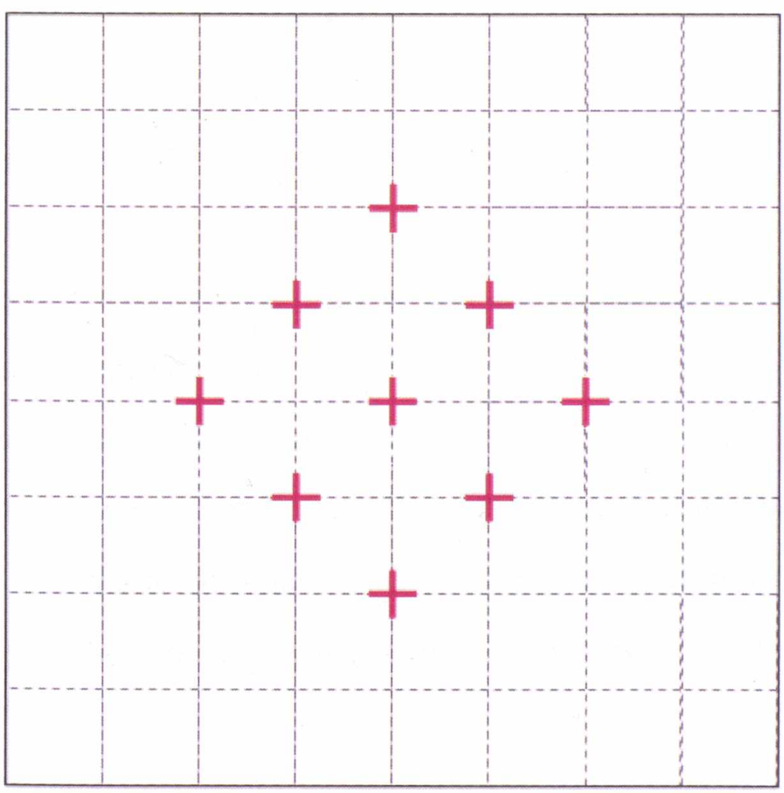

Figure 2.4: Pattern for diamond search.

2.1.2.9 Motion Compensation

This method of motion compensates the current image in order to minimise the energy of the residual. In that part one will discuss about the most common motion compensation techniques. Firstly it would be the block motion compensation, then, the overlapped block motion compensation and finally specific sub-pixel motion compensation know as half-pel motion compensation.

2.1.2.10 Block Motion Compensation (BMC)

In this compensation technique, one divided the frame into blocks. The current block is used to predict the follow one. The only modification made on the next block is to give it to become the reference frame for the following block. In other words, the best matching block predicted on the reference picture is used as the predicted best matching block for the following picture. The good point about this technique, is that the result is actually very accurate, but the main problem, is that the block motion compensation creates some horizontal and vertical edges. These edges are generated by a discontinuity at the border of the block.

2.1.2.11 Overlapped Block Motion Compensation (OBMC)

To avoid the problem of the block borders discontinuity, one can use the Overlapped block motion compensation (OBMC). In that compensation, the size block is double the previous size. Then each block overlaps quadrant-wise with it eight nearest blocks. In this architecture each pixel is included in 4 different blocks, which means each pixel has been predicted by 4 blocks.

2.1.2.12 Half-Pel Motion Compensation

It is a special in the Sub-Pixel Motion Compensation (SPMC) as shown in Figure 2.5. This method consists of using a prediction from interpolated sample positions in the reference image based on the fact that the best estimation can be found by using fractional pixel accuracy [49]. Interpolated pixels are computed between the real pixels of the image and the estimation is carried out upon those new pixels. The result is really improving but the working process become really heavy.

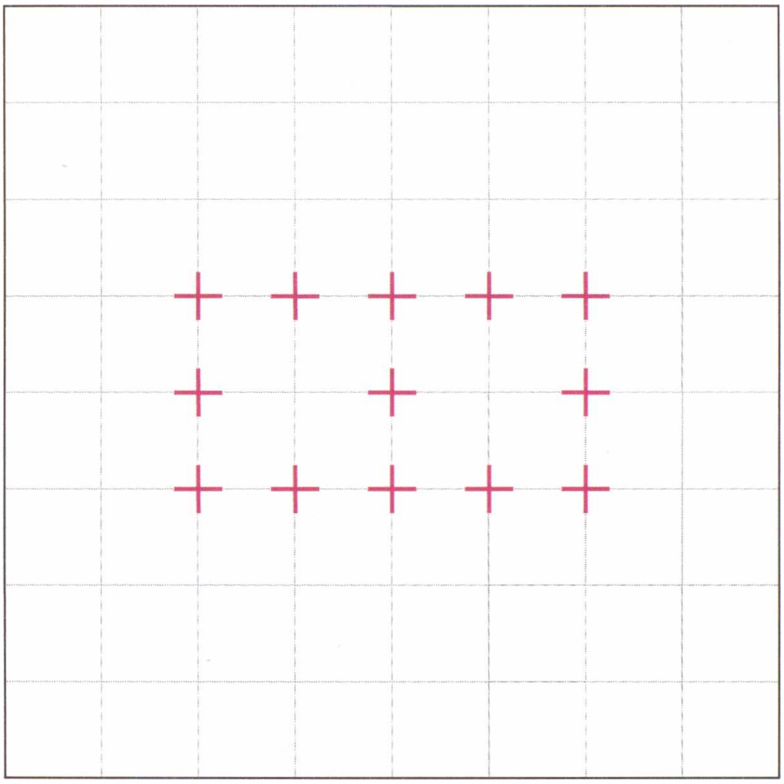

Figure 2.5: Half-pel motion compensation

2.1.3 Principle of Spatial Redundancy Compression

To reduce the space redundancy, one will process the residual frame from the previous block. A transform coding is necessary to change the frame to another domain, this process must follow some specific criteria [52]:

- the data in the new domain should be decorrelated and compact

- the process should be reversible

- the process should be computationally tractable.

Two transforms are well-known when speaking about video compression: a block based transform – the Discrete Cosine Transform (DCT) – and an image-based transform – the Discrete Wavelet Transform (DWT). One will briefly discuss these two methods in this chapter. A more detailed development of discrete wavelet transform will be found later in the thesis.

2.1.3.1 Discrete Cosine Transform (DCT)

The discrete cosine transform is an orthogonal transform that changes a frame F(x,y) into frequencies F(u,v). If one is looking at the correlation between near pixels from an 8x8 block, one will find the point sprayed but follow a straight line. This is due to the fact that two pixels near each other are slightly different but not much.

The main goal of this frequency transformation is to decorrelate the pixels, that will regroup the information of the frame on only few frequency parameters. In the transformation, the highest values of the coefficient are for the high frequencies which represent the shape of the picture and the low frequencies have lower values coefficient and contain textures. Those coefficients are called AC and DC, DC coefficient is the lowest frequency DCT coefficient and equals the average of all the pixels in the block. All other coefficients are AC coefficients. The compression takes place at that point. The DC coefficient will be compressed on a smaller amount of bits. The decimal value will be truncated to get a full value. This type of coding scheme is called the entropy coding, it is mean the number of bit allocated to each composant is related to the quantity of information contained.

One can explain this transform mathematically, if one gets a pixels matrix {A:B}. If one apply the transition matrix P to {A:B}, one get 8 orthogonal vectors. One then move to a new base at 8 dimensions. The parameters of the matrix P follow the equation:

$$Pij = Cj\sqrt{\frac{2}{N}}.\cos\left(\frac{(2i+1)j\pi}{2N}\right) \tag{2.5}$$

The mathematical equation of the new system can be written as:

$$F(u,v) = \frac{2[C(u).C(v)]}{N}\sum_{x=0}^{N-1}\sum_{y=0}^{N-1}F(x,y).\cos\left[\frac{\pi}{N}.u\left(x+\frac{1}{2}\right)\right].\cos\left[\frac{\pi}{N}.v\left(y+\frac{1}{2}\right)\right] \tag{2.6}$$

Where $C_{(v=0)} = \sqrt{\frac{1}{2}}$, $C_{(v=0)} = \sqrt{\frac{1}{2}}$ otherwise $C_{(u>0)} = 1$, $C_{(v>0)} = 1$

2.1.3.2 Discrete Wavelet Transform (DWT)

The basic operation in wavelet transform is to filter an image with a low pass filter and a high pass filter and down-sample the output by a factor of two. One then obtain two new images, Low and High. Those pictures follow the same process (filtered and down sampled), but in the y direction. One have now four sub-band images, in order to recover the original frame, one can easily recombine them. One still have the same amount of information but this new architecture will be more appropriate for efficient coding. One developed deeply the discrete wavelet transform in section 2.1.4.

2.1.3.3 Quantisation

Even after the transformation, one still has the same volume of data and the whole image is still carried, one has not yet performed a compression. However, those manipulations have been made to facilitate the distribution of the energy which is easier to reduce. For example in the DCT coefficients, most of the energy has little value because the lowest frequency components concentrate the maximum of energy. The quantisation will remove the non important values. The principle of quantisation is to divide the values by a quantisation value (positive integer) and approximate the quotient to the nearest integer. Two features can define the uniform quantises, the step Q and the factor S. As the human eye is less sensitive to distortions at high frequencies, one can have a more efficient result by applying larger step size quantisation [52].

One of the most popular quantiser is the variable uniform quantiser. The principle of this method is to apply by a quantising lookup table which provides quantisation step sizes and multipliers that scale the coefficients to smaller values [53].

Let us consider a N by N matrix of coefficient to be quantised, in order to have a different position for each values of the parameter q, the step size q is determined from N by N matrices.

To ensure good entropy coding of those quantised coefficients, a reorder should be perform. This reorder will is explained just at the next section.

2.1.3.4 Reordering

The Zigzag scanning is the most common way of reordering the coefficients of a block. It allows an array of data to be obtained with the non-zero coefficients regrouped at the beginning and a sequence of zeros at the end.

1	2	6	7	15	16	28	29
3	5	8	14	17	27	30	43
4	9	13	18	26	31	42	44
10	12	19	25	32	41	45	54
11	20	24	33	40	46	53	55
21	23	34	39	47	52	56	61
22	35	38	48	51	57	60	62
39	37	49	50	58	59	63	64

Table 2.1: Order in which the coefficients are scanned for reordering in the zig-zag scanning.

2.1.4 Discrete Wavelet Transform

This section follows the work done by Muriel Castro Dufourny in June 2004 [54], herself following the work made by Ghanbari [55]. He uses to explain the wavelet transform by firstly explaining subbands and proceed to the description of discrete wavelet transform and by giving some examples of common wavelet transforms.

2.1.4.1 Sub-band

The idea is to separate the signal in frequency domain into two sub-bands (high-band and low-band) so that each band can be down-sampled for transmission and then up-sampled and combined for reconstruction of the original signal. When the image is filtered by cut high and low frequencies, each resulting sub-band can be sub-sampled.

For a better result at the reconstruction part, an ideal filter should be implemented, unfortunately it is not possible and degradations have been introduced by the filters. To avoid the degradation introduced by component, at the decoder after up-sampling and filtering, the bands are summed again. One can by designing the filter in such a way that the degradation from the down sampling filters will be cancelled by the degradation of the up sampling filters, in that case a very good reconstruction can be expected.

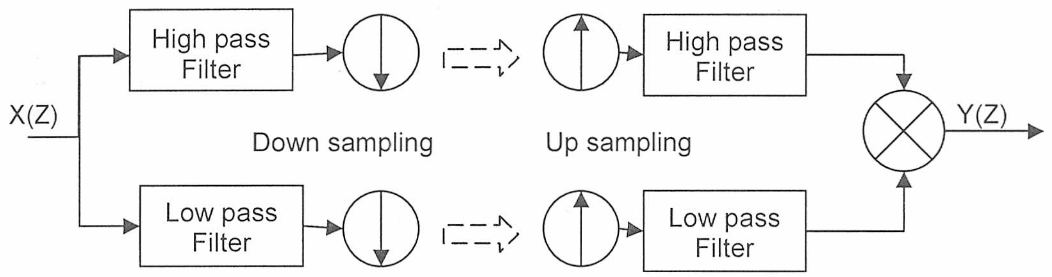

Figure 2.6: Sub-band block diagram.

In order to obtain a perfect reconstruction on Figure 2.6, one need to obtain X(Z) = Y(Z), but this will introduce delay to proceed the filters and the down and up sampling.

2.1.4.2 Wavelet transform and Discrete Wavelet Transform

Wavelet is based on the approximation theory, in contrast with the subband coding which is based on frequency analysis. This theory is based on the Fourier expansion, which explained that a signal can be expressed as the sum of a series of sines and cosines. Only Frequency resolution is used in the Fourier analysis, when in the wavelets, both time and frequency variation will be recreated [55].

- **Definition**

The mathematical expression of the wavelet transform is:

$$X_w(a,b) = \frac{1}{\sqrt{a}} \int_{-\infty}^{+\infty} x(t)\Psi_{a,b}(t)dt \tag{2.7}$$

where a mother wavelet Ψ is dilated to the scale parameter *a* and translated at the position parameter *b* to form the basis function Ψ defined by:

$$\Psi_{a,b}(t) = \Psi\left(\frac{t-b}{a}\right) \tag{2.8}$$

If you have a one dimension function and you transform it through the wavelet transform, you will get a two dimensional function. In the same idea, if you transform a two dimensional function through a wavelet transform you will get a four dimensional function. For images, the discrete wavelet transform is used basing the approach on the fact that any square integral function *x(t)* can be represented as a linear combination of functions as explain in the book [54]:

$$x(t) = \sum_{m=-\infty}^{+\infty} \sum_{n=-\infty}^{=\infty} \alpha_{mn} \Psi_{m,n}(t) \tag{2.9}$$

where α_{mn} are the wavelet transform coefficients:

$$\alpha_{m,n} = \int_{-\infty}^{+\infty} x(t) \Psi_{m,n}(t) dt \tag{2.10}$$

In other words, the wavelet transform will be a set of filters with coefficients equivalent to discrete wavelet functions. The concept of higher order systems is used to describe a more complex version by adding stages for the decomposition.

2.1.4.3 Standard wavelets

The wavelet function Ψ and the scaling function φ, shown on the previous section, are the best to describe standard wavelet transform. Usually the number of vanishing moment can be represented a number next to the wavelet number. Two important properties of wavelets are the admissibility and the regularity conditions. One may need some other characteristics to design wavelet transforms as it explained on [56] [57].

However one will now trying to explain some well known and very used type of wavelet. Some typical wavelets and their properties will be listed here.

2.1.4.4 Haar wavelet

The Haar wavelet is one of the most commonly used wavelets. It resembles a step function and is defined by:

$\Psi(t) = 1$ if $0 \leq t < 0.5$

$\Psi(t) = -1$ if $0.5 \leq t < 1$

$\Psi(t) = 0$ otherwise (2.11)

$\varphi(t) = 1$ if $0.5 \leq t < 1$

$\varphi(t) = 0$ otherwise

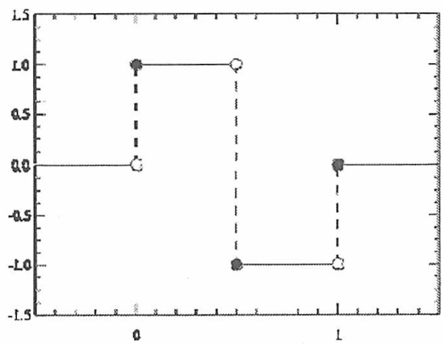

Figure 2.7: The Haar wavelet representation [58].

2.1.4.5 Daubechies wavelets

The Daubechies wavelet is a big family of wavelet (Haar is part of this group). These wavelets have no explicit expression. They are orthogonal and have compact support. The properties of symmetry or the number of vanishing moments depend on the order N of the dB.

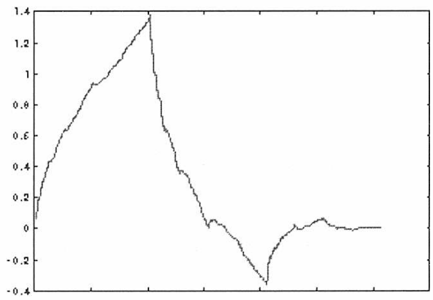

Figure 2.8: The Daubechies wavelet order 3 [59].

2.1.4.6 Shannon wavelet

The Shannon wavelet is used as the base of multiresolution approximation as follow:

$$\Psi(t) = \frac{\sin 2\pi(t-1/2)}{2\pi(t-1/2} - \frac{\sin \pi(t-1/2)}{\pi(t-1/2)} \quad (2.12)$$

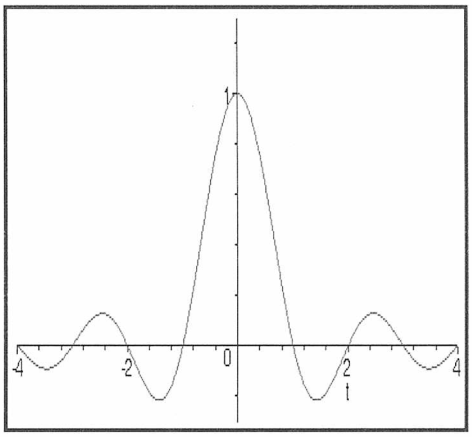

Figure 2.9: The Shannon wavelet [60].

2.1.4.7 Mexican hat wavelet

The definition of the Mexican hat wavelet is:

$$\Psi(t) = \frac{1}{\sqrt{2\pi\sigma^{3}}} \cdot \left(1 - \frac{t}{\sigma}^{2}\right) \cdot e^{\frac{-t^{2}}{2\sigma^{2}}} \tag{2.13}$$

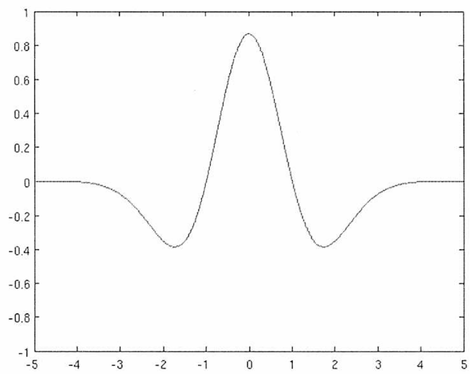

Figure 2.10: The Mexican hat wavelet [61].

Many other wavelets can be described. Several examples are available in the two books referenced as [55] [57].

2.1.4.8 Usage and utilisation

Wavelets are heavily used in signal processing as well as frequencies detection and noise reduction. When dealing with picture, they can be used again for noise reduction, but also for object recognition (wavelet-based semi-automatic segmentation, motion compensation, contrast enhancement, texture analysis, and of course, compression as developed in the books [55] [62] [63] [64] [65].

2.1.4.9 Compression with wavelets

In Figure 2.1 the transform block has now been changed from DCT to DWT. For image compression, the most commonly used wavelets are Daubechies wavelets. This wavelet will introduce some changes in the steps of quantisation and variable length coding. If during the transformation the level of decomposition used is too low, some remaining correlation should appear between pixels. That lack of level of decomposition can be reduced by different pulse code modulation coding. The higher-order wavelet coefficients are quantised by successive approximation and afterwards by exploitation of similarities of the bands. They will then be Embedded Zero Tree Wavelet (EZW) encoded described here [54]. Here are the following steps for a compression with wavelets:

- Compute mean of image.

- A zero-mean image is reached by the application of an R-stage wavelet.

- An initial yardstick will have a length *l* of half the maximum absolute value of the wavelet coefficients.

- Three lists are generated, one determines the coordinates of the coefficient in the order they will be scanned, and two others are subordinate and temporary lists.

- The wavelet coefficients are scanned and assigned '0' if they are smaller than the current yardstick length, $\pm\frac{3l}{2}$ otherwise.

- Following the order described in the dominant list, the reconstructed coefficients are scanned again, generating a string of symbols: the sign of a coefficient is appended to the string, and its coordinates to the subordinate list. If a coefficient is zero, its coordinates are appended to the temporary list [54].

- The yardstick length is now set to $l/2$.

- The coefficients which previously have not been reconstructed as zero are scanned again according to their order in the subordinate list, and adding to them either +l/2 or -l/2 and the corresponding sign is appended. The subordinate list is reordered so that the coefficients whose reconstructed values have higher magnitudes come first. The '+' and '-' symbols of this pass are encoded with an arithmetic coder.

- The temporary list is erased after replacing the dominant list. The whole process can be repeated several times until the size of the bit stream is reached. A header will include information necessary to the decoder; the number of stages in the wavelet transform, image dimensions, initial value of the yardstick l, image mean etc [54].

2.1.5 Principle of the Entropy Coding Compression

At this stage, one is coding the quantised transformed coefficients in order to reduce the bit rate. Entropy encoding encodes the data according to the information content. The length of each message is coded with the shortest code when the probability of occurring is high, while the lowest probability of occurring are coded with the longest code. By following this method, any Variable Length Coding (VLC) algorithm can be used but the Huffman coding and the Arithmetic coding are the most used and one are then going to describe them.

In the Huffman algorithm, one is trying to reach the shortest average possible code word length. A variable length coding has been affected to every symbol based on their probability of occurrence. A Huffman code tree is then created with the calculated probabilities of each symbol. This tree follows the steps:

-order the data in increasing order of probability.

-sum the two lowest probabilities and assign the value to the new combination.

-restart the process including in the list of data the probability of the new combination.

These three steps are repeated until one single join remains. Then, one is passing through the branches of the tree by starting from the last join to finish on each data. Each branch is labelled '0' or '1'. Then by following the path, one can generate a code by the sequence of '0' and '1' encountered.

The data encoded by the Huffman coding has a very accurate representation to the original data. But the main disadvantage in this method is the tree, as it has to be transmitted to the decoder in order for it to follow the same path; this introduces more data to be transmitted. This method for huge video sequence will introduce some delay for the computation. To avoid this problem a pre-calculated variable length coding can be used as the arithmetic coding.

Arithmetic coding provides a probability, 0 to 1, and assigns an interval to every symbol. The occurring frequency of each interval in the message determines the size of the interval itself. The higher the probability, the larger the range assigned.

The main problem in this method is the whole codeword has to be received before being decoded, and if a bit is corrupted the whole message can be corrupted.

After the entropy coding, the process of compression is nearly done. The motion vectors, VLC-quantised coded coefficients and additional information like parameters; algorithms used are defined and will then be multiplexed in order to generate a bit stream. One will then in the next section introduce the standard one used during this project.

2.2. MPEG-4

MPEG-4 is an ISO/IEC standard developed by MPEG (Moving Picture Experts Group). Hundreds of researchers around the world contributed to the evolution of this standard. ISO/IEC 14496 (the formal designation of MPEG-4), was finalised in October 1998 and became an International Standard at the beginning of 1999. The fully backward-compatible extensions under the title of MPEG-4 Version 2 were frozen at the end of 1999, to acquire formal International Standard status in 2000. Several extensions have been added since and work on some specific items is still in progress. MPEG-4 is now successfully used in digital television, interactive graphics applications and in interactive multimedia (Internet, wireless transmission). MPEG-4 can be represented as a structure, as shown in Figure 2.11:

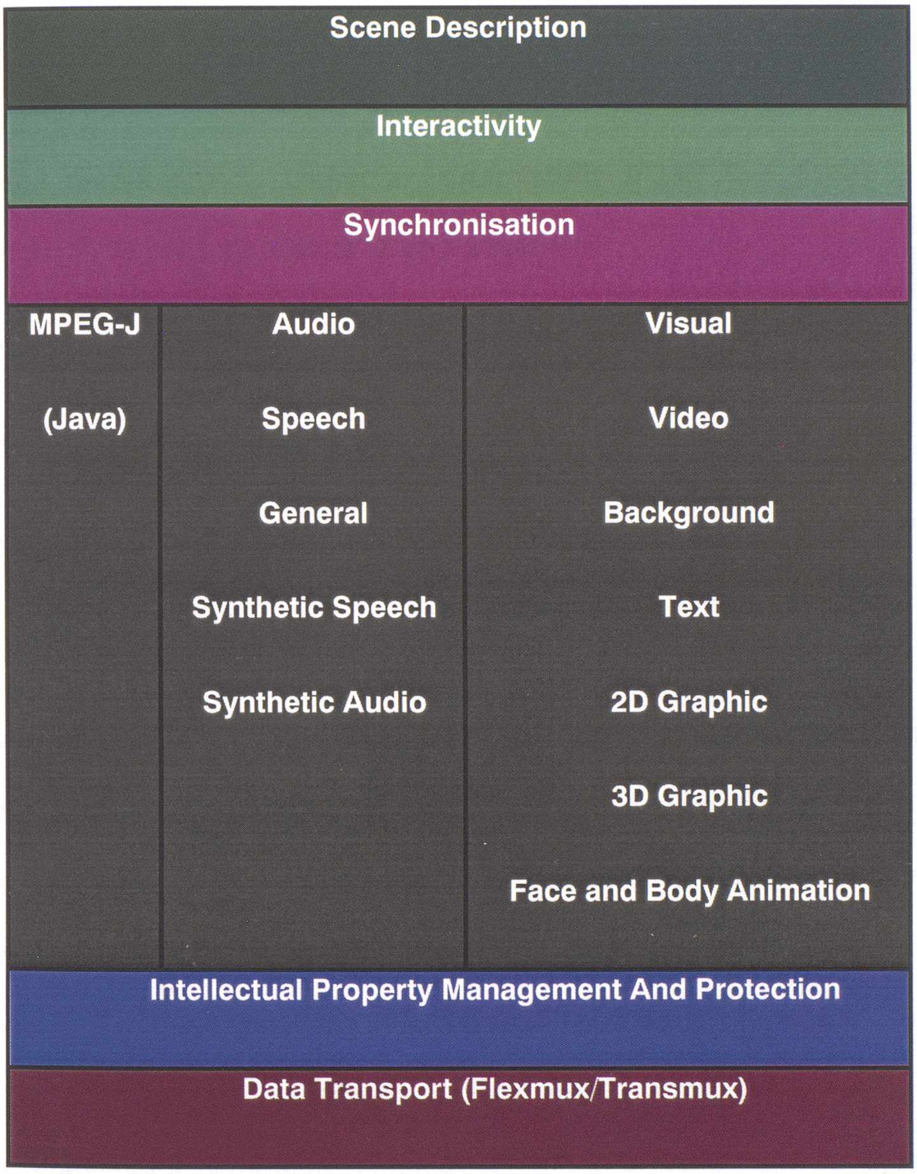

Figure 2.11: The MPEG-4 Structure

2.2.1 Description of the MPEG-4 standard

Several media objects compose hierarchically a MPEG-4 scene. One can highlight three primary objects listed as below:

- A fixed background

- Video objects (e.g. a talking person)

- Audio objects (e.g. a voice, background music)

In order to handle an audiovisual scene, one can use media object. This media object in its coded form is composed of descriptive elements. In this coded form objects can be consider individually from each other. During a transmission, media objects can send by streaming within one or multiple elementary streams. To deal with the hierarchically-encoded information, an object descriptor is used. It associates for one media object all the streams. This object descriptor contains the object content information and all the copyright properties.

In order to follow the same parameters, each stream has information on the required decoder resources and the precision of encoded timing information (it can as well include information about Quality of Service).

The network provides the synchronisation of streaming information from source to destination, if it has different QoS. A multiplexer with two layers is used to deal with the delivery layer and the synchronisation layer.

The Delivery Multimedia Integration Framework (DMIF) specification manages the first layer of the multiplexing. In order to reduce the multiplexing by grouping into elementary streams, one can include this multiplex into the MPEG-defined FlewMux tool. By doing this multiplexing at this level can reduce the delay by limiting the network connections. The layer which deals with matching of the quality of service is modelised by the Transport Multiplexing layer (TransMux layer). The MPEG-4 does not deal with the data packets mapping or with the control signalling, as it is only specifies the interface to this layer.

Different industries have been working with MPEG to create the way to identify intellectual property in media object content. The intellectual property control systems have been implemented inside the MPEG-4 code sequences at the systems layer level. Few possibilities has been created, the most secured is by storing a number issued by an international numbering system. This number will be a unique identifiers number applied to identify the rights of a media object. The problem is that some source does not have this number, then MPEG-4 decided to offer the possibility of generating a Key value pair to identify intellectual property.

2.2.2 Functionalities

Advanced algorithms have been developed to compress video and audio data, to create these algorithms a toolbox has been created in MPEG-4. The data has been completely rebuilt at the receiving end as the encoded data streams can be transmitted or memorised independently.

The description of the scene constituted by a relationship between audio and video contents has been addressed by the systems part of the MPEG-4. Two main levels can describe the relationship:

- The arrangements of the objects in a scene can be described by a spatiotempral sequence that one call Binary Format for Scenes (BIFS). The users can interact with the content by modifying the order of the objects in the scene for example.

- Another level is the relationship between the elementary streams. It is defined by the Object Descriptors. It is for example to define the audio and the video of a videoconference participant. Some information is carried out by the Object Descriptors such as the intellectual properties, the decoders characteristic needed to describe the elementary streams and the URL of theses elementary streams.

A hybrid coding has been used at the MPEG-4 Visual standard. This coding can mix natural images and video coding with computer generated scenes. At the receiving end, the visual standard is composed by both natural image and video coding tools and the characteristic of the synthetic compression algorithms.

MPEG-4 visual can supports:

- bit rates: typically between 5 Kbit/s and more than 1 Gbit/s

- Formats: progressive and interlaced video

- Resolutions: typically from sub-QCIF to high level of resolutions (4k x 4k pixels).

In these very efficient algorithms, one can compress the coding with an adjustable quality of texture. This quality can be adjusted from a very low factor of compression (almost no loss) up to a very high ratio of compression (acceptable level for the user). Many tools can be used to address the error resiliency aspects of a wireless access, another tool is the video compression algorithms in error-prone environments.

2.2.3 Technical description of MPEG-4

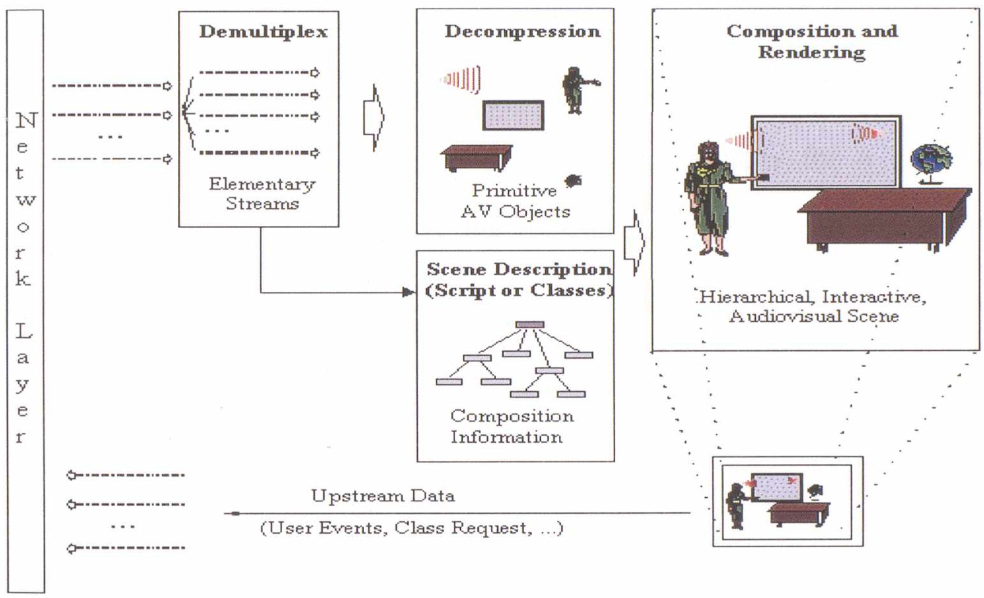

Figure 2.12: Major components of an MPEG-4 terminal (receiver side) [66]

Figure 2.12 shows how stream is demultiplexing by passing through the FlexMux demultiplexers that recover the elementary streams. The appropriate decoder has been use to decompose and to pass the Elementary Streams to the next action. This last action is to generate the render on the appropriate device by recovers the AV objects from the media object encoded form and reconstruct the original AV object by the right reconstruction operations.

The systems decoder model let the encoder to select and memorise in a buffer the minimum of data required to decode a sequence, it is the way to allow

MPEG-4 to predict how the decoder will behave when it decodes the various elementary data streams. During the set up of a MPEG-4 sequence, the buffer resources are transmitted inside the object description to the decoder. This let the decoder decided if it is possible to handle the session. By deciding the buffer availability; it is possible to stop sending information if the receiver and has not space left to receive. As this information is accessible at anytime, it is then possible to choose the amount of channel's capacity for a real time encoding. Furthermore the stream of the data transmitted has to enclose the timing information.

One has two types of timing information:

- The first is used to give time base to the decoder.

- The second contains the composition and expiration time for composition units.

With this timing information, at the decoder one can adapt the inter-picture interval and audio sample rate in order to equal the encoder's inter-picture interval and audio sample rate for the operation of synchronisation. Systems operation without any timing information is allowed, defining a buffering model is not possible for this case.

2.2.4 Description of MPEG-4 Video

The description on this part is only based on natural origin visual object. It is possible in MPEG-4 visual standard to allow efficient storage, manipulation of the texture... The video objects are the representation and the decoding of very small units of video content. To illustrate, one can have a first video object as a car (no background) then one add another video object music, to create a complete scene. The visual part of the MPEG-4 standard provides solutions in the form of tools and algorithms for:

- Efficient compression of images and video

- Efficient compression of textures for texture mapping on 2-D and 3-D meshes

- Efficient compression of implicit 2-D meshes

- Efficient compression of time-varying geometry streams that animate meshes

- Efficient random access to all types of visual objects

- Extended manipulation functionality for images and video sequences

- Content-based coding of images and video

- Content-based scalability of textures, images and video

- Spatial, temporal and quality scalability

- Error robustness and resilience in error-prone environments

MPEG-4 Video supports conventional rectangular images and video as well as images and video of arbitrary shape. This concept is illustrated in Figure 2.13.

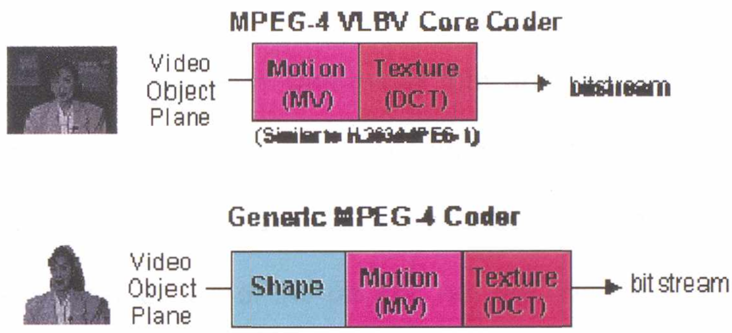

Figure 2.13: The Very Low Bit-rate Video Core and the Generic MPEG-4 Coder [67].

The MPEG-4 is similar to the conventional MPEG-1 and 2 for the coding of conventional images and video. It is using motion prediction and motion compensation before encoding the texture. For the content-based functionalities, one added at the algorithm a coding shape and transparency information. An eight bit transparency component can be used to represent the shape, that is let the description of the transparency possible event if the video object is made of other objects. Spatial and temporal scalability are highly supported by the MPEG-4 standard. Theses two scalability are using a classical rectangular shape or an arbitrary shape. Scalability refers to the ability to decode only a part of a bit stream and reconstruct image sequences with:

- reduced decoder complexity and thus reduced quality

- reduced spatial resolution

- reduced temporal resolution.

MPEG-4 has an error resilience parameter that one can divide in three main categories, resynchronisation, error concealment and data recovery. This category has been use as well in many other codec. The length of the video is based on the number of bits of the packet, not the number of macroblocks. A new video is created at the beginning a new macroblock if the number of bits included in the actual media exceeds he predetermine threshold.

Data recovery system tries to recover the information, usually lost, just after the new synchronisation. Theses techniques are encoding the information in an error-resilient manner not as a simple error correcting codes. For the Reversible Variable Length Codes (RVLC), the variable length codeword is designed such that it can be read both in the forward and the reverse direction. An example illustrating the use of a RVLC is given in Figure 2.14. With a standard coding scheme, all the data would be lost between the two synchronisation. However, by using a RVLC, one is able to recover some data by following the process in the next figure. It should be noted that the parameters QP and HEC shown in the Figure 2.14 represent the fields reserved in the video packet header for the quantisation parameter and the header extension code, respectively.

Re-synchronisation Marker	MB Add.	QP	HEC	Forward → Decode	Errors	Backward ← Decode	Re-synchronisation Maker

Figure 2.14 : Reversible Variable Length Code

In any video codec, error concealment is very important in error management. The efficiency of the error strategy scheme is deeply dependent on the quality of the resynchronisation strategy. Fundamentally, to have better error concealment, the resynchronisation technique has to locate the error. The actual resynchronisation method is very efficient for low bit-rate system and small delay application with the implantation of simple concealment strategy (duplicating blocks from previous frame).

A new synchronisation has been made in order to separate the motion and the texture data. This new information of synchronisation has been made between the motion and the texture data. In order to conceal an error of missing texture, the design uses the motion information. By following the error, the previous decoded video object plan has been motion compensate by the actual motion. A new algorithm has been developed by the MPEG teams, this technique provide in real time error recovery (called NEW PREDiction (NEWPRED)). Between the decoder and the encoder the system employs an upstream channel and, hence to the error conditions, the encoder changes the reference frame. This technique use only predictive frame and no intraframe, the efficiency of the system have been confirmed even in very hard condition.

- Burst Error on the wireless networks (around 10^{-3}average)

- Packet Loss on the internet (around 5%).

The MPEG-4 coding scheme support the content-based functionalities, as it has a very good result on arbitrary shape representation. This standard support nearly all the functionalities of the previous version of MPEG coding system (MPEG-1 or 2)like for example, frame rates, bit rates, temporal and quality scalability, etc...Figure 2.15, a fundamental representation of the bit rate video core and the functionalities available on the MPEG-4 standard.

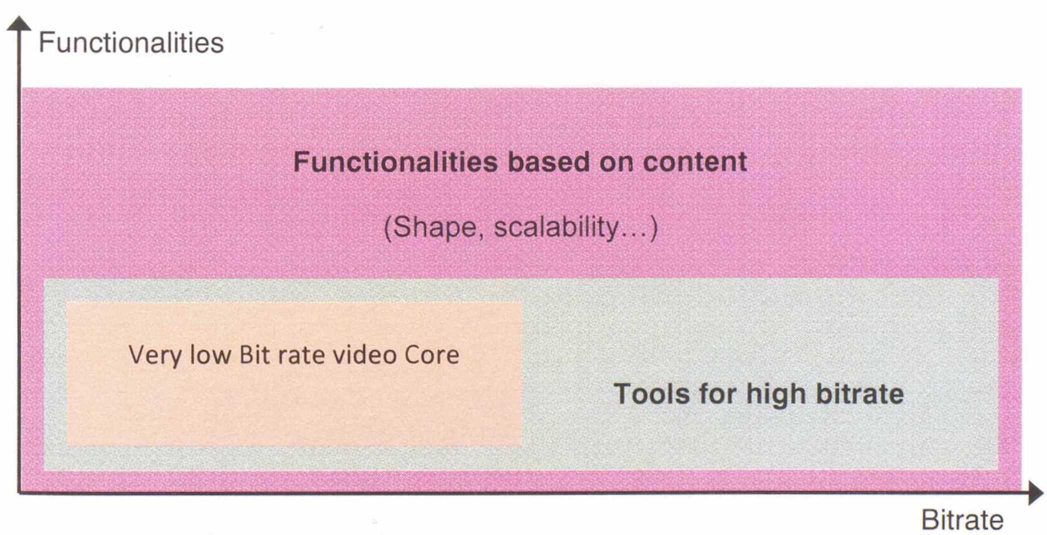

Figure 2.15 : A basic classification of the bit rates and functionalities.

For rate from 5 Kbits/s to 64 Kbits/s, the very low bit-rate video core provides algorithms, for frame rates up to 15Hz and a low spatial resolution picture. The application above 64 Kbits/s up to 10 Mbits/s, regroup the multimedia broadcast with a quality as good as the digital TV.

2.2.5 The MPEG-4 Video Image Coding Scheme

A basic overview of the MPEG-4 video algorithms is shown on the following figure. It is the dlock diagram to encode rectangular as well as arbitrarily shaped input image sequences.

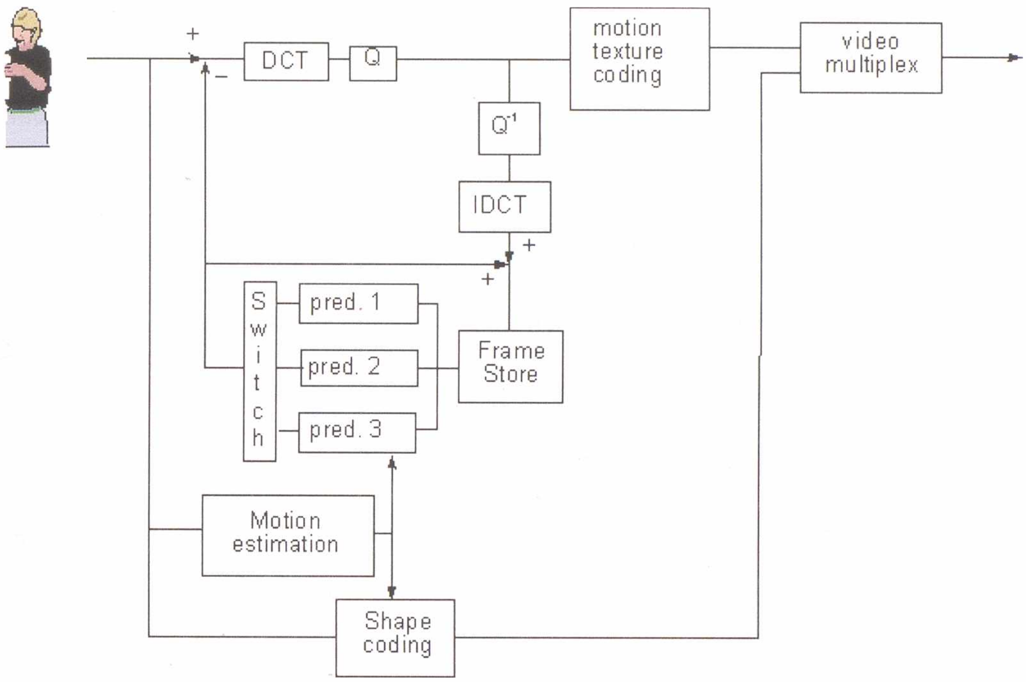

Figure 2.16 : Basic block diagram of MPEG-4 Video Coder [68].

A shape coding, a DCT based texture coding and motion compensation are the base of the coding algorithm. In order to reach an effectiveness compression of some video sequence, the standard uses some tools based on the object based motion prediction. One has available different methods of motion prediction solutions to increase the coding efficiency and a more flexible representation of the object, such as:

- Standard 8x8 or 16x16 pixel block-based motion estimation and compensation, with up to Quarter per Motion accuracy.

- Global Motion Compensation for video objects: The object is encoded with a very small amount of parameters. This technique is using global motion estimation, image warping, motion trajectory coding, and texture coding for prediction errors. Global motion compensation based on a possibly large still image (background). Only few global motion parameters are coded to be able to reconstruct the object from consecutives frames. To obtain a better precision for the motion compensation, Quarter per Motion Compensation is used. This method is very light in computational usage and slightly easy to program. With a correct motion description, one gets a better quality of picture of the video sequences.

- Shape-adaptive DCT: It is based on predefined orthonormal sets of one-dimensional DCT basis functions. The concept of the MPEG-4 standard is shown on the Figure 2.17. The idea is to say that the background (tennis court and crowd, picture top left on Figure 2.17) can be separated from the foreground object (here the player, picture top right). The background has been then sent only once to the receiver and then only the relevant camera parameters are transmitted to the receiving end. Then the receiver recomposes the complete frame by aggregating the background and the foreground together (bottom of Figure 2.17.

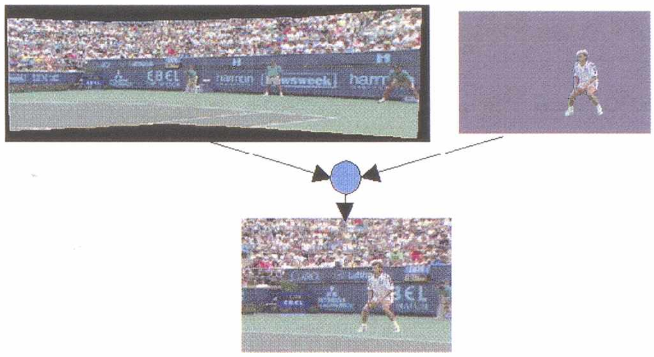

Figure 2.17: Example of sprite coding of video sequence [69].

2. 3. IEEE 802.15.1

The security side is provided by the IEEE 802.15.1 standard and Bluetooth technology. In the IEEE 802.15.1 standard, the Work Group has included a Service Access Point (SAP) [70]. SAP is uniquely identified by an address. In order to make two adjacent layers to communicate, some rules must be set up for the interface. An entity from the layer *N+1* gives an entity of the layer *N* an Interface Data Unit (IDU) through the SAP. The IDU is actually made up of two things; a Service Data Unit (SDU) and some Interface Control Information (ICI). The SDU is the data that two peer entities exchange, but it is also what the layer *N+1* of the sender gives to the layer *N*. The control information is used to help the lower layer. It is, for example, the number of bytes contained in the SDU (this can be used in the function that controls data integrity). To send a SDU, a layer *N* might need to split it into several parts. When exchanging this data with its peer entity, every piece of data is fitted with a Protocol Control Information (PCI) contained in a header, and then sent as a Protocol Data Unit (PDU). Headers are used by peer entities to carry their peer protocol. This PDU then becomes the SDU for the layer *N* and is going to be sent to the layer *N-1* via the SAP.

IEEE 802.15.1 and Bluetooth technology uses a spread-spectrum technique called frequency hopping rather than a transmitter over a single Industrial-Scientific-Medical (ISM) frequency band. IEEE 802.15.1 and Bluetooth standard operates in the 2.4 GHz band (unlicensed frequency). The 2.4 GHz frequencies are the same range as that used by the wireless networking standard IEEE 802.11. In comparison, Bluetooth and IEEE 802.15.1 have a 1Mbit/s theoretical bandwidth, and are lower powered than 802.11, which has 11Mbit/s of bandwidth. This means it has shorter data packets, and faster frequency hopping [71].

The data rate in IEEE 802.15.1 and Bluetooth wireless technology is given by three types of data links between two different devices. These three types are as follows:

- Asynchronous type: 723Kb/s in transmission (54.6Kb/s in reception)

- Synchronous type: 432Kb/s (reception and transmission)

- Voice/data type: 64Kb/s (reception and transmission)

A number of devices use this frequency, like microwave ovens, cordless telephones and baby alarms. These all produce unpredictable interferences to IEEE 802.15.1 and Bluetooth devices. The frequency hopping of IEEE 802.15.1 and Bluetooth must therefore transmit data in successive small packets on different frequencies. This enables interference to be avoided on a particular fixed frequency, and data packet loss to be minimised. IEEE 802.15.1 and Bluetooth devices use discrete 1 MHz channels within the 2.4 GHz band resulting in 79 distinct carrier frequencies (RF channels). Two Bluetooth and IEEE 802.15.1 devices connected to each other must use a common frequency-hopping sequence to ensure that they stay synchronised. One of ten different hopping sequences can be chosen and, once the devices are initially synchronised, they can jump to a new frequency and still communicate. Bluetooth and IEEE 802.15.1 have 1600 frequency hops per second and a 625 ms transmission period at a particular frequency.

To permit coexistence between Bluetooth and other wireless technology, Adaptive Hopping is being developed, which detects and avoids any potential problems.

2.3.1 Antennas

This type of device requires antennas, which radiate in patterns as close to a sphere as possible, so that they can connect in any direction, at any angle. For devices such as LAN access points, a more directional antenna could be beneficial so that coverage could be restricted to a single room. Different antenna types will radiate in different patterns, so the type of product may affect the antenna choice. The available space and the cost of different types of antennas will also affect the choice of antenna for a product. Surrounding components and casing can affect the performance of an antenna as well. In particular, the feed and proximity of ground planes can affect an antenna's characteristics. Thus, the positioning and the characteristics of the antenna must be considered when designing the product.

2.3.2 Radio Interface

The 802.15.1 devices operate at 2.4 GHz in the globally available, license-free ISM band. Although widely available, the use of this band by many other systems and the pollution of it by other sources such as microwave ovens make this a rather hostile environment. 802.15.1 employs a fast frequency-hopping scheme to counteract this, together with error protection and correction. The specification of the radio requirements has been relaxed as far as possible to facilitate low cost and low power design. The modulation used is simply GFSK with one symbol per bit, providing a gross bit rate of 1 Mb/s from a channel bandwidth of 1 MHz. By specifying tight constraints on symbol timing and drift rate, the task of recovering the received data stream is also simplified. In spite of this, there are some key design issues to be addressed in making such a low cost system reliable and robust, and some close attention must be given to system design. Other parts of the 802.15.1 specification itself impose some constraints on the radio system and these must also be accounted for. To facilitate the re-use and interoperability of radio parts with different base band devices, an initiative named blueRF has been launched to standardise the 802.15.1 radio interface.

2.3.3 The Link Controller

The link controller layer is responsible for managing device discoverability, establishing connections, and once connected, maintaining the various on-air links. It does this through a set of state machines, which drive the base band through the following stages to establish links.

- Host request inquiry

- Inquiry is sent using the inquiry hopping sequence.

- Inquiry scanning devices respond to the inquiry scan with FHS packets which contain all the information needed to connect with them.

- The contents of the FHS packets are passed back to the host.

- The host requests connection to one of the devices that responded to the inquiry.

- Paging is used to initiate a connection with the selected device.

- If the selected device is page scanning it responds to the page.

- If the page-scanning device accepts the connection it will begin hopping using the master's frequency-hopping sequence and timing.

The control of the links rests with the local device. Although the master governs when devices may transmit once connected, it is up a device to allow itself to be connected with. If a device does not make itself discoverable by inquiry scanning, it cannot be found, in a connection, it is free to disconnect without warning at any time. Once a base band link is established the master and slave can exchange roles, so that slave becomes master and master becomes slave. The base band links are used to carry link management traffic, voice traffic and data traffic. Mechanisms are also included to support multiple piconets and the exchange of the master and slave roles between communicating devices, together with several low power periodic operating modes. The link control and base band work very closely together and careful consideration must be given to overall system design and partitioning for an optimal solution.

2.3.4 The Link Manager

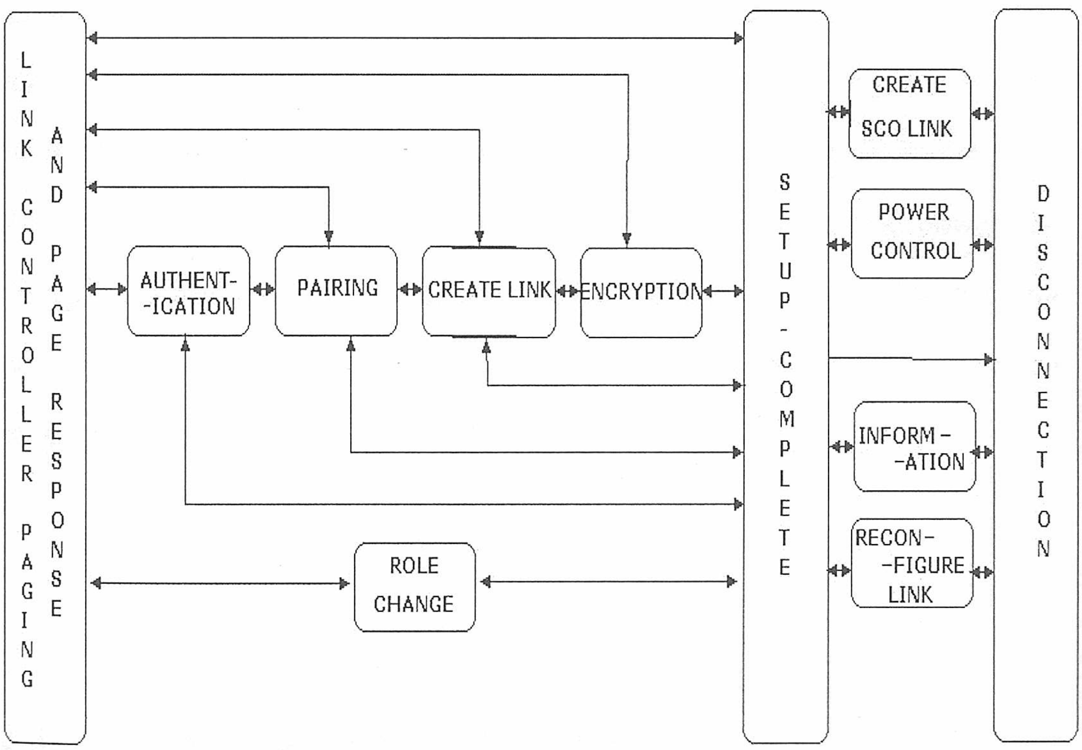

Figure 2.18: The link manager

As Figure 2.18 shows, there are many alternative paths through this diagram. The simplest route after the Link Controller has finished with paging and page response is to just exchange messages to confirm connection setup is complete. More complex routes can involve changing the roles of master and slave, either after paging, but before the connection setup is complete, or at any time thereafter. Similarly, the security procedure's authentication, pairing, and encryption can be passed through on the way to connection setup-complete, or they can be visited at any time afterwards. Once the connection has been set up, it can have up to three Synchronous Connection-Oriented connections created across it, or its mode can be changed, either to a low mode or test mode. The link can be also be reconfigured at any time, including at mode changes, quality of service changes, packet types changes, and power level changes. Finally, information about an active link can be received at any time. When the connection is no longer required, the link manager protocol can cause disconnection.

2.3.5 The Host Controller Interface

An 802.15.1 device can be made up of two parts; a host implementing the higher layers of the stack, and a module implementing the lower layers. The Host Controller Interface (HCI) provides a uniform interface between an 802.15.1 host and its module. Because the Host Controller Interface is standardised, host software can work with 802.15.1 devices from a variety of manufacturers. The Host Controller Interface uses three packet types:

- Commands which go from host to module,

- Events which go from module to host,

- Data packets which travel in both directions.

Between them, these three packet types can be used to completely control an 802.15.1 module and to transfer any data required. The Host Controller Interface commands allow the host to completely control an 802.15.1 module, including: control of links, including setting up; tearing down and configuring links. Setting the links depends on whether power saving modes and role switches are allowed.

Direct access to information on the local 802.15.1 module, and access to information on remote devices, is gained by triggering the link manager protocol exchanges.

- Control of many base band features such as timeout.

- Retrieving status information on a module.

- Invoking 802.15.1's test modes for factory testing, and for 802.15.1 qualification.

2.3.6 The Logical Link Control and Adaptation Protocol

The Logical Link Control and Adaptation Protocol (L2CAP) provides the facilities higher layer protocols need to communicate across an 802.15.1 link.

The facilities are:

- Establishing links across underlying Asynchronous Connectionless channels using L2CAP signals.

- Multiplexing between different higher layer entities by assigning each one its own connection ID.

- Providing segmentation and reassembly facilities to allow large packets to be sent across 802.15.1 connections.

Higher layer applications and protocols communicate with the L2CAP using implementation depend primitives known as L2CAP signals. The standard defines what these signals should do, but the exact implementation can vary from one 802.15.1 system to the next. All applications must use L2CAP to send data. It is also used by 802.15.1's higher layers such as RFCOMM and Service Discovery Protocol, so L2CAP is an essential part of any 802.15.1 System.

2.3.7 Encryption and Security

IEEE 802.15.1 has powerful security features with the SAFER+ encryptions engine using up to 128 Kbits. As the link level, it is possible to authenticate a device: This verifies that a pair of devices share a secret key derived from 802.15.1 passkey, also known as a Personal Identification Number (PIN). The 802.15.1 passkey is either entered in a user interface, or for devices which do not have a user interface, it can be built in by the manufacturer. After authentication, devices can create shared link keys which can be used to encrypt traffic on a link. The combination of authentication and creating link keys is called pairing. At the application level, pairing, possibly accompanied by exchange of higher level security information, is called bonding. Authentication may be repeated after pairing, in which case the link key is used as the shared secret key. Three modes of security can be implemented.

- Mode 1 is not secure.

- Mode 2 has security imposed at the request of applications and services.

- Mode 3 has security imposed when any new connection is established.

An 802.15.1 security white paper suggests architecture for implanting security in the higher layers of an 802.15.1 protocol stack. This is based on Mode 2 security. In addition to being authenticated by the link management procedures, the security architecture introduces the idea of devices being authorised by a user to use particular services. The security architecture suggests implementing this through a pair of databases: one holds information on which devices are authenticated and/or authorised, and the other holds information on whether services require authentication, authorisation and/or encryption. Services and protocols registration, the central security manager, grants permissions to use services. Security is essential to many applications which use 802.15.1 links, but hiding the complexity of 802.15.1 security from the user is essential if 802.15.1 devices are to be easy to use. Through the security architecture, it is possible to implement security at a variety of levels with minimal intervention from the user.

2.3.8 Low-power Operation

Many 802.15.1 devices will be operated by batteries, so it is important that they do not use more power than necessary. Some 802.15.1 devices, such as headsets connected to cellular mobile phones, need to respond fast to signals, so ideally they should stay connected all the time to avoid the delay of setting up a connection extending their response time. However, being constantly connected would mean using the radio a lot and would drain the device's batteries. IEEE 802.15.1 provides three low-power modes which extend battery life by reducing activity on a connection. These modes are:

- PARK: Park mode provides the greatest opportunities for power saving. The device only wakes up in periodic beacon slots, where it listens for unpark transmissions from the master. If it is not unparked, it goes back to sleep, switching off its receiver. A special unpark message is used to restore the device to normal activity. Devices which are parked give up their active member addresses, so the unpark message either uses a special parked member address, which is assigned to devices when they are parked, or can use the device's 802.15.1 Device Address (BD_ADDR). Because a parked device gives up its active member address, one master can have more than seven devices in park mode.

- SNIFF: In Sniff mode, the device wakes up periodically and listens for transmissions, but no special unpark messages are needed to communicate with it. Devices in sniff mode keep their active member address. Typically, sniffing devices will be active more often than parked devices.

- HOLD: Hold mode just puts a connection in a low-power state for a single period. Connections to slave might usefully be put into Hold mode while a master performs an inquiry, as the master knows in advance that it will not be able to service the connections for a while.

2.3.9 Quality of Service

For the IEEE 802.15.1 specification, Quality of Service (QoS) means attaining the required data rate, delay variance and reliability. The link manager provides QoS capacities by a variety of means, including choosing packets types, setting polling intervals, allocating buffers, allocating bandwidth to links and deciding whether or not to perform scans. Link managers negotiate peer to peer to ensure that QoS is coordinated at both ends of a link. At the L2CA level too, QoS is negotiated end to end. One L2CA signals its peer to request QoS this peer in turn asks its local link manager if it can fulfil the request. On systems with HCI, this interaction between L2CA and link manger is accomplished through a series of HCI commands and events. Alternatively, with dot to dot link, a reliable link is provided by the receiver acknowledging packets. Any packets which are not acknowledged are retransmitted. Broadcast packets are not acknowledged so, to provide a reliable link, an IEEE 802.15.1 device can be set to retransmit broadcast packets a certain number of times.

Sometimes on point-to-point links, the base band's attempts to provide a reliable link can result in data being kept in the memory and retransmitted time and time again while sending, so a washing-out mechanism can be used to override the reliable link and allow the system to put data away. This can be done directly with a command, or automatically with a timer. L2CA and the lower layers of the stack do their best to provide the QoS which higher layer applications and protocols request, but as with any wireless system, there are never any real guarantees, so sometimes the QoS requested cannot be achieved. In such cases, the lower layers tell the higher layers by sending QOS_Violation events, so the higher layers of the stack know the required QoS is not being achieved and can decide what to do about it.

CHAPTER 3:

Artificial Intelligence Techniques

3.1 Neural Network

3.1.1 Introduction

The toolbox of engineers is having a new signal-processing technology tool, the artificial neural network. This field is used in many areas from medical to aviation. We will provide a brief overview of the theory of the neural network, learning rules, and applications of the most common models. Adaptability is the key world of the Artificial Neural Network (ANN), which is used to learn from a non-linear system in order to perform a function from set of data. As the system is learning during the process, it changes the system parameters during the execution, which one calls the training phase. As soon as the training is completed, the parameters of the ANN model are fixed and the system starts running to resolve the problem. The system is based on rules that the model follows step by step to match the constraint of the system and / or to optimise the criteria. In neural network algorithms the most important part of the design is the training of input/output data. These information carry the essential record to define the optimal operating point. As this learning method has a nonlinear structure, it gives flexibility to the system that lets the process deal with almost any type of problems. This flexibility and adaptability is worth developing a bit more.

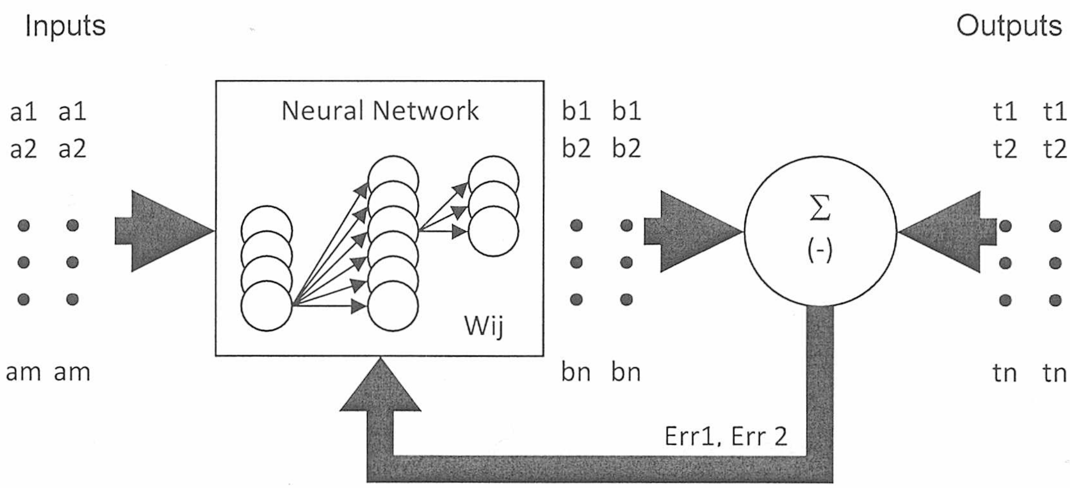

Figure 3.1: Example of neural network computation.

As one can see on the Figure 3.1, an input (a1, a2...) is entered in the neural network model, together with the target response set (b1, b2...) at the output of the neural network. This desired value is then subtracted from the output in order to have this error between these two values. This error is then sent again to the neural network model to adapt and adjust the parameters following the rules previously implemented to optimise the output until the result is acceptable. One can now easily understand the tremendous importance of the input and output data in those types of algorithms.

One can highlight a major difference between the traditional engineering system and those new designs. Indeed, the "classic" systems are set up at the beginning and the process runs always in the same way. Now with the learning system, the system adapts its parameters automatically. The designer specifies the topology, the learning rules, the criterion to stop the training phase and some other parameters.

ANN is nowadays one of the best solutions for difficult problems which are very difficult to deal with correctly with “classic'” technologies. ANN-based systems provide now a very efficient response in terms of time of development and resources. Denker said that "artificial neural networks are the second best way to implement a solution" motivated by the simplicity of their design and because of their universality, only shadowed by the traditional design obtained by studying the physics of the problem [72]. Now the artificial neural network is developed in many areas such as voice recognition, system identification, pattern recognition and many more....

3.1.2 The natural model

Artificial neural networks emerged after the introduction of simplified neurons by McCulloch and Pitts in 1943 [73]. To understand the artificial neural networks one need to understand the natural neural network. Here is a brief explanation of the biological model.

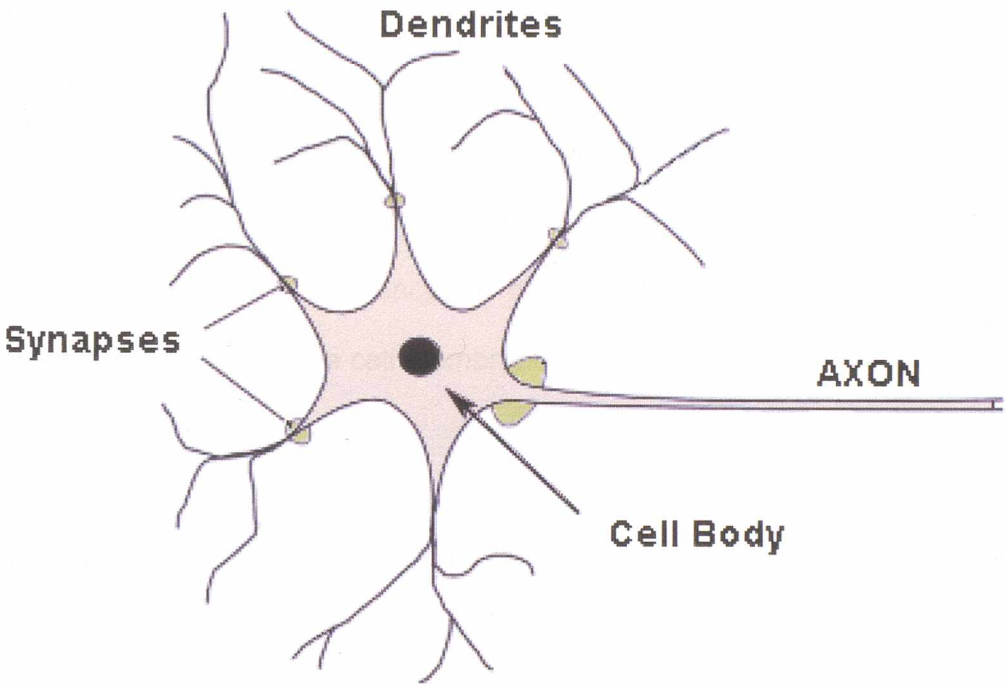

Figure 3.2: Simplified natural neurons

As one seen in the Figure 3.2, a natural neural is composed of four main parts. The cell body has two contiguous parts, the dendrites and the axon. At the middle of the cell one find the cell body, inside the cell body one have the nucleus and the maintaining protein synthesis. The "branches" of the cell is called the dendrites, each cell can have few dentrites which receives signals sent from other neurons. The Axon is a growing part of the cell body, usually only one axon is present on each neuron. The action of the axon is to transmit the electric signal generated at the pre-synaptic terminals. The electric signals are called action potentials, and they are used to convey information to the brain.

The last main part of a neuron is the synapse, it is the area of contact between two neurons. The neurons are connected by the synaptic cleft. These synaptic clefs let the electrical signal going through the neuron by thirteen different chemical interactions. The signals are generated by the membrane potential, which is based on the differences in concentration of sodium and potassium ions inside and outside the cell membrane.

Neurons can be classified into three separate categories. The first group provides the data for the discernment and the motor synchronization, this is called sensory neurons. The second category is called motor neurons and it is responsible for sending information to muscles and glands. The inter-neuronal group is the last category and regroups all the rest of the neurons.

3.1.3 The artificial model

To design an efficient artificial model of a natural neuron, one needs to follow three important components. Firstly the synapses have to be considered as the weights. The value of the weight models the strength of the connection between the neurons and the input. An excitatory connection would be a positive value of the weight, while a negative weight would reflect an inhibitory connection [74]. The neural cell activity will be modelled by the two other components. After being modified by the respective weights, the inputs will be combined. Finally, the amplitude of the message at the output of the neuron is controlled by an activation function; the acceptable value for the output is commonly between -1 and 1 (or 0-1).

Mathematically, this process can be shown as in Figure 3.3.

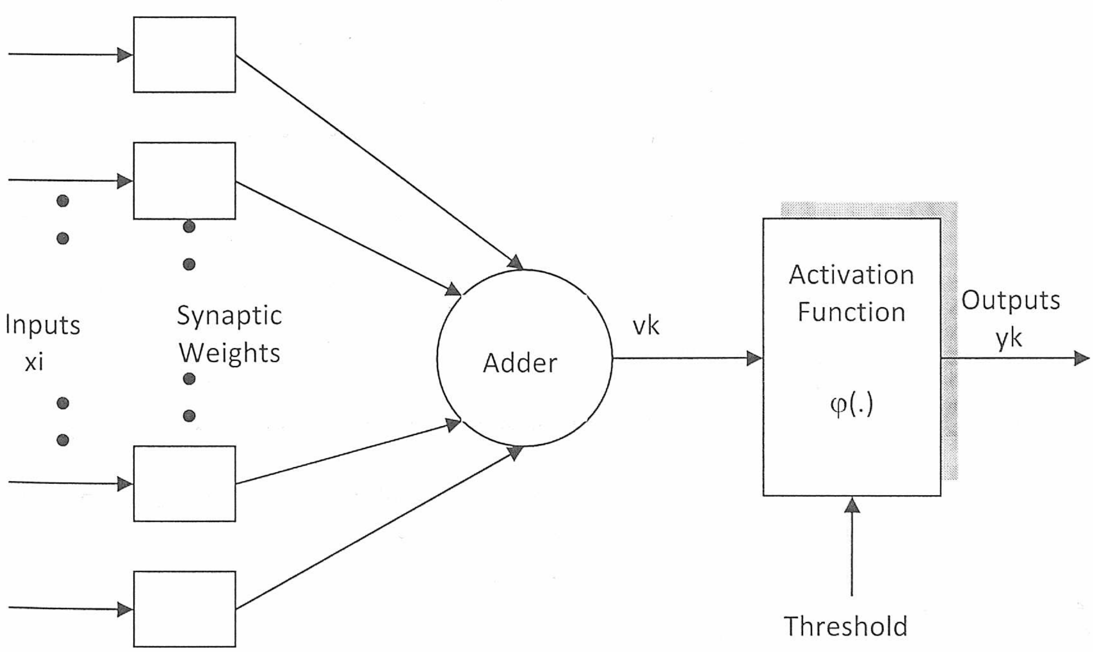

Figure 3.3: The mathematical model of a neural network

The activity of a neuron can be modelled mathematically by the following equation:

$$v_k = \sum_{j=1}^{p} w_{kj} x_j \qquad (3.1)$$

The output of the neuron, yk, would therefore be the outcome of some activation function on the value of vk.

3.1.4 Activation functions

The activation function, like previously said, proceeds like a limitator function as the value of the output of a neuron is usually between -1 and 1. Three activation functions commonly symbolised by φ(.) are used. First, if the value of the summed input is under a specific threshold value (v) then the threshold function has the value 0. And the threshold function gets the value 1 if the sum of the input is more or equal to specific (v) value.

$$\varphi(v) = 1 \text{ if } v \geq 0 \qquad (3.2)$$

$$\varphi(v) = 0 \text{ if } v < 0$$

The second function is called Piecewise-Linear function. This function can, as shown on the equation 3.3, get the values 0 and 1. But depending on the amplification factor, the function can get also values between 0 and 1.

$$\varphi(v) = \begin{cases} 1 & v \geq \frac{1}{2} \\ v & -\frac{1}{2} > v > \frac{1}{2} \\ 0 & v \leq \frac{1}{2} \end{cases} \tag{3.3}$$

Finally, one has the sigmoid function. This function can range -1 and 1, as well as 0 and 1. Here one has an example of a sigmoid function, the hyperbolic tangent function. In Figure 3.4, one has a representation of some distributed function.

$$\varphi(v) = \tanh\left(\frac{v}{2}\right) = \frac{1-\exp(-v)}{1+\exp(-v)} \tag{3.4}$$

The Parallel Distributed Processing (PDP) is the original base of the artificial neural network that one is using. To perform the processing, one is using very similar neural network architecture.

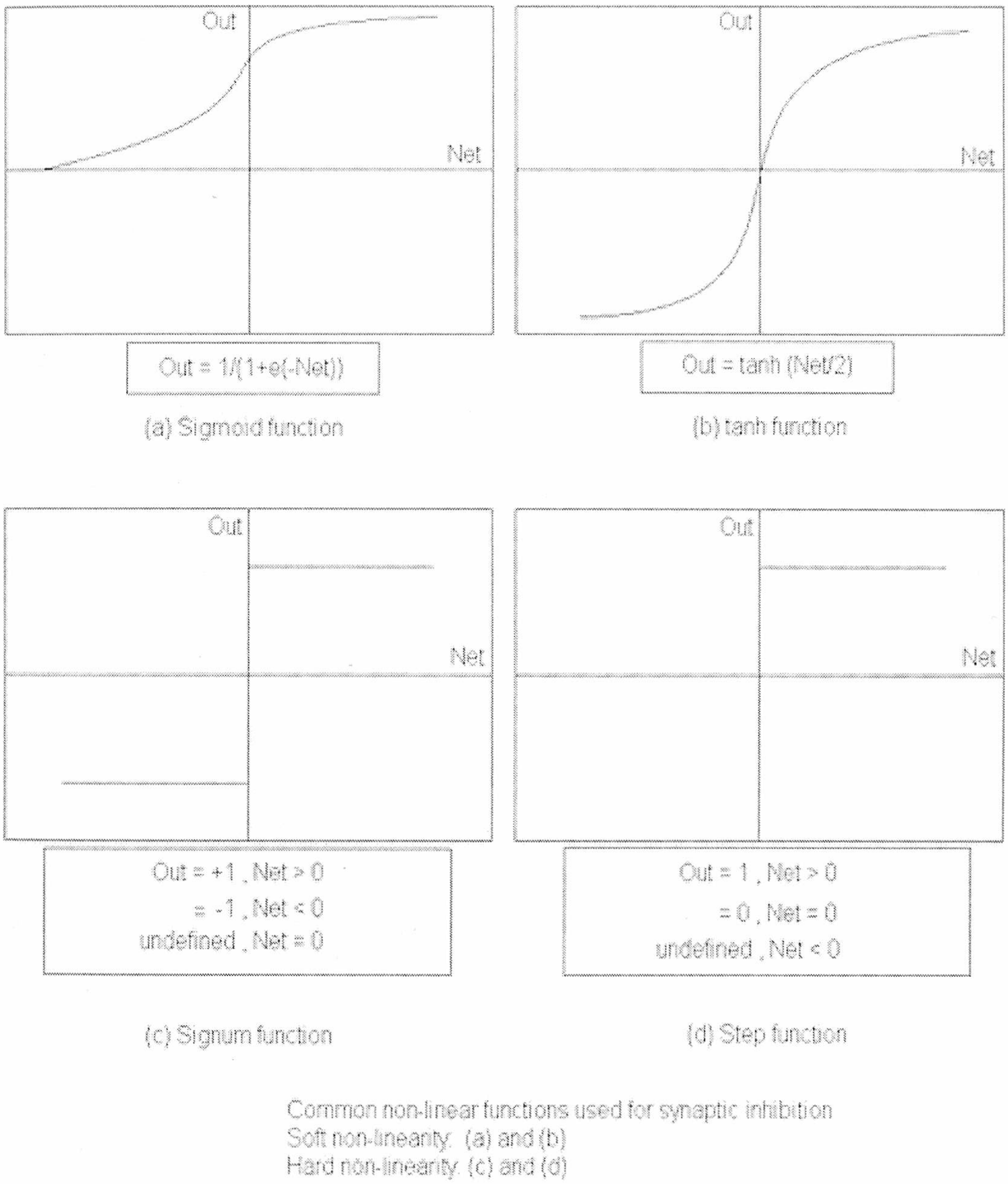

Figure 3.4: A distributed representation

To simplify an artificial neural network uses a large number of weighted connections to communicate to each other by sending a group of simple processing units. The following part summarise the major aspects of a parallel distributed model [75]:

- A set of processing units ("neurons", "cells");

- A state of activation *yk* for every unit, which is equivalent to the output of the unit;

- Connections between the units. Generally each connection is defined by a weight *wjk* which determines the effect which the signal of unit *j* has on unit *k*;

- A propagation rule, which determines the effective input *sk* of a unit from its external inputs;

- An activation function *fk*, which determines the new level of activation based on the effective input *sk(t)* and the current activation *yk(t)*;

- An external input *Φk* for each unit;

- A method for information gathering;

- An environment within which the system must operate, providing input signals and error signals.

3.1.5 Processing units

The action of a unit is to use the input received to transmit a proceeded output to the other units. The unit has to as well deal with the weights adjustment, as the system can be seen as a parallel communication system. In fact few units can send messages at the same time. At this level one need to highlight three different types of units; the first unit receives the data from the outside of the network and it is known as the input units. The output of the network has the output unit which sends the data at the exterior of the network. And the last unit is called the hidden unit which keeps the input and output inside the network. Synchronous or asynchronous mode can be used to modify the criterion during the system is proceeding. The main difference between these two methods is: with the synchronous method the update of the activation is made at the same time, while with the asynchronous mode the update can appear only on every specific time *t*.

3.1.6 Neural Network topologies

This section will talk about the topology of an artificial neural network, where the topology is the design of the communication involving the propagation of data and the units. One has to choose between feed-forward and the recurrent patterns. Here is a brief explanation of each of these two patterns:

On the feed-forward neural networks, no feedback is available; the data is going forward from the input to the output units. Where the data from input to output units is strictly feed-forward. Multiple units can be used to extend the data processing; it allows the connection to extend from the outputs to the inputs of the units in the same layer. Classical examples of feed-forward neural networks are the Perceptron and Adaline [76].

On the recurrent neural network, feedback is possible and an important point in here is the dynamic properties of the design. Sometimes, the neural network has to move to a stable level, and with the recurrent neural network the output activation value will keep unchanged, while with other pattern, model will keep changing significantly as the dynamic performance constitute the output of the system [77]. Examples of recurrent networks have been presented in 1982 and 1988 [78] [79].

3.1.7 Training of Artificial Neural Networks

To obtain outputs requested, the inputs have to produce the desired values that the neural network is configured for. In order to set the strengths of the connection, it is possible to explicit the set the weight by known information. The other method is to let the system learn by itself, following defined rules and to let it train the weights to optimise the system response.

Learning situations can be supervised or unsupervised. In the supervised method, input and output patterns train the neural network, the input/output can come from the system itself or from an external source.

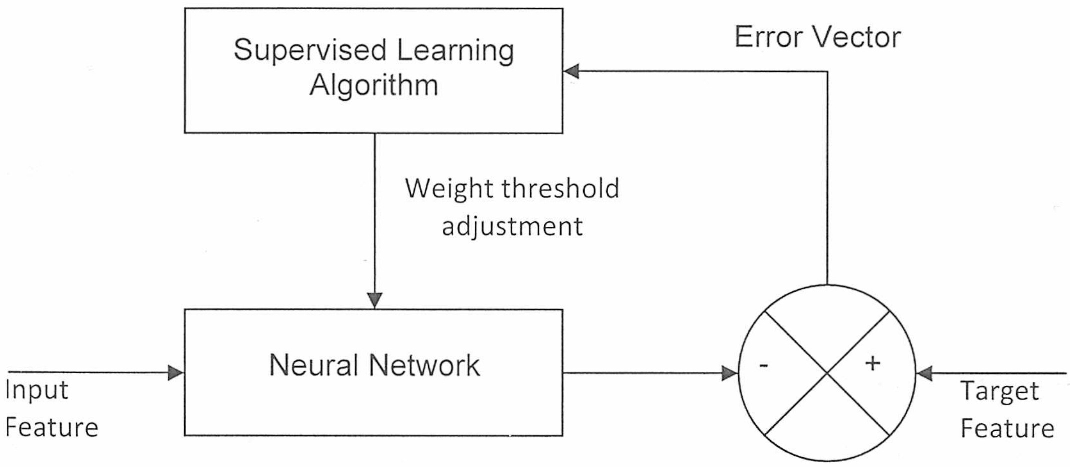

Figure 3.5: Diagram of the training of artificial neural networks

One puts at the input of an unsupervised learning system cluster of patterns to train the output. As the system has no defined rules, the model has to find the relevant features of the input units. The system has to develop its own representation of the input.

A third type of learning can be placed between these two previous types. The reinforcement learning receives some feedback from the environment after it did some action. The adjustment of the parameters will be made accordingly to the environment's feedback. The system will continue to change the parameters until a stable state has been reached, at that moment no more modification of the parameters will be made.

3.1.8 Modifying patterns of connectivity of Neural Networks

Following defined rules, the unsupervised and supervised learning method adjust the weight of the units' connections. In 1949 Hebb proposed the hebbian learning model [80], the two models one presented earlier can be considered as variants of Hebb's model. The idea is that if two units *j* and *k* are active together, their interconnection can be strengthened. Considering *k* the input of *j*, the weight w_{jk} can be modified by the formula of the simplest version of Hebbian learning:

$$\Delta w_{jk} = \gamma(y_j . y_k) \tag{3.5}$$

where γ is a positive constant of proportionality representing the learning rate.

Another common rule adjusts the weights by using the difference between the actual and desired activation, as shown on the following equation:

$$\Delta w_{jk} = \gamma y_j (d_k - y_k) \tag{3.6}$$

in which d_k is the desired activation provided by a teacher.

3.2 Fuzzy Logic

The first development of the idea of fuzzy logic as a possibility to process data, not as now as a control methodology, came from the University of California at Berkeley by Prof. Lofti Zadeh [81]. This idea was to give membership rules to processing data instead of non membership data.

It was during the 1970s, that fuzzy logic was applied to control systems. This late application of this technology to the control system was due to a lack of power of the computer before that date.

The concept is to admit that people do not need precise values at the input, as they are capable of very adaptive control. Indeed, if inaccurate input and noise can be tolerated by the feedback controllers of a system, the answer must be more efficient and simple to realize.

This new concept of fuzzy logic applied to control system, can be used for a wide range of purposes, from the simplest implanted controller to the largest multi-channel data acquisition workstation. This methodology can be applied on software and hardware side of a process. When the input of a system is not precise, noisy, has a high probability of missing data at the entrance, a very simple way to determine an unambiguous end is to use a fuzzy logic design. To easily the concept of fuzzy logic, one can compare the response of the system to the reaction of a person, but that decision for the fuzzy logic would be faster than the human.

By opposition of the mathematical way to resolve a problem, fuzzy logic is dealing only with simple rule-based (IF AND THEN). To design a fuzzy logic model, the most important aspect is the understanding of the difficulty by the designer rather than his technical knowledge. For example, rather than dealing with rate control in terms such as:

Actual rate = 600Kbps, maximum rate <724Kbps	Or	700Kbps< actual rate <800Kbps

One can use:

IF (rate under the value) AND (rate is getting lower) THEN (allow more information transmitted)	Or	IF (rate is high) AND (rate getting higher) THEN (reduce amount of information transmitted)

These terms are imprecise and yet very descriptive of what must actually happen. Consider what you do in a transmission if the rate level is low: you will allow more data to be transmitted. Fuzzy logic is capable of imitating this type of performance but at very high rate.

An exact value for a fuzzy system is not really required, but it is indispensable to have some numerical parameters in order to estimate the error and to analyse the evolution during the process. For example, a simple rate control system could use a single rate feedback controller whose data is subtracted from the command signal to compute "error". Error might have units of Kbps and a small error considered to be less than 20Kbps while a large error is more than 500Kbps. The error evolution might then have units of Kbps/second with a small error evolution being less than 5Kbps/s and a large one being more than 100Kbps/s. These values do not have to be symmetrical and can be "adjusted" once the system is operating in order to optimise performance. Generally, fuzzy logic is so forgiving that the system will probably work the first time without any adjustment.

The problem can be described as follow:

IF	500 Kbps	AND	5 Kbps	Then	Very small adjustement
	20 Kbps		5 Kbps		Small adjustement
	500 Kbps		100 Kbps		Medium adjustement
	20 Kbps		100 Kbps		Big adjustement

Table 2.2: Example of a “IF – AND – THEN” fuzzy rules.

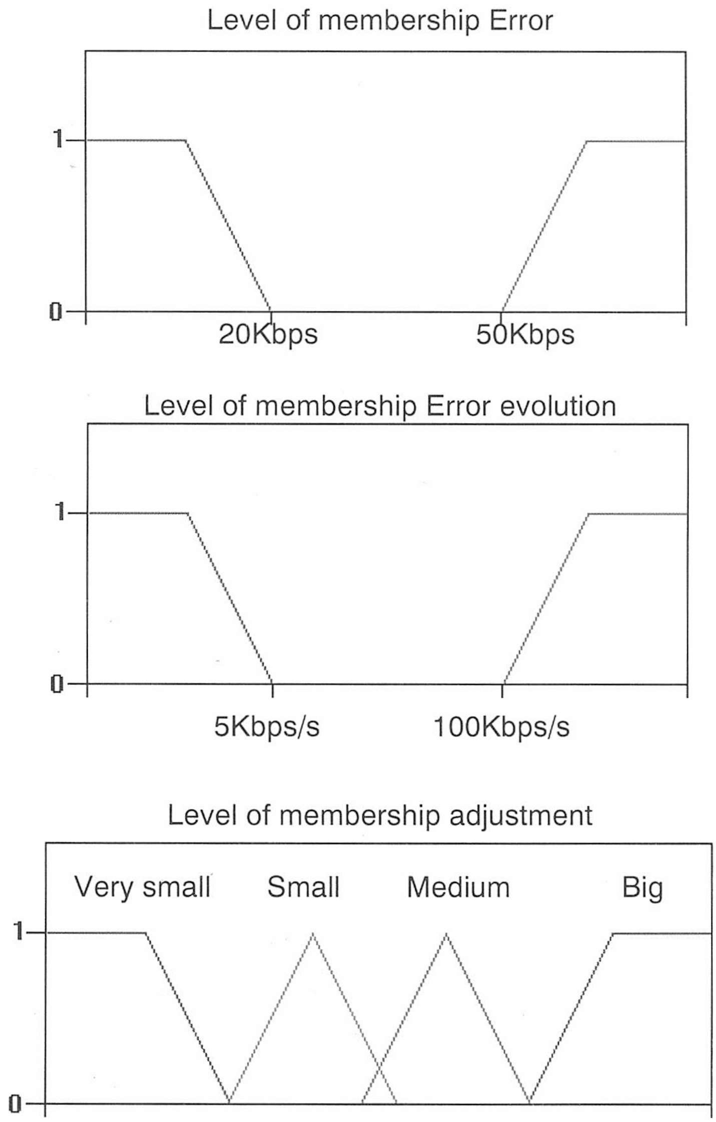

Figure 3.6: Example of membership for a fuzzy system

Fuzzy logic was initially created to categorise and treat data, but research proved that it is also an excellent method for many control system applications, as this method tries to imitate the human being logic. Another positive point about this concept is its flexibility as it can be applied from small to very large processes and from the simplest to the most complex system. A fuzzy logic algorithm is very robust and tolerant of the data input as well as operator. It can be implemented in a very hard atmosphere like a very noisy area, because it is using imprecision to adapt to the right target.

CHAPTER 4:

Development of intelligent approach to MPEG-4 transmission over IEEE 802.15.1

In this chapter, a presentation of the design will be made. In the Figure 4.0; one has an overview of the complete process of the system. The chapter will explain each block in depth in order to get a full understanding of the hybrid neural and fuzzy system. The following block diagram explains visually the way this design works:

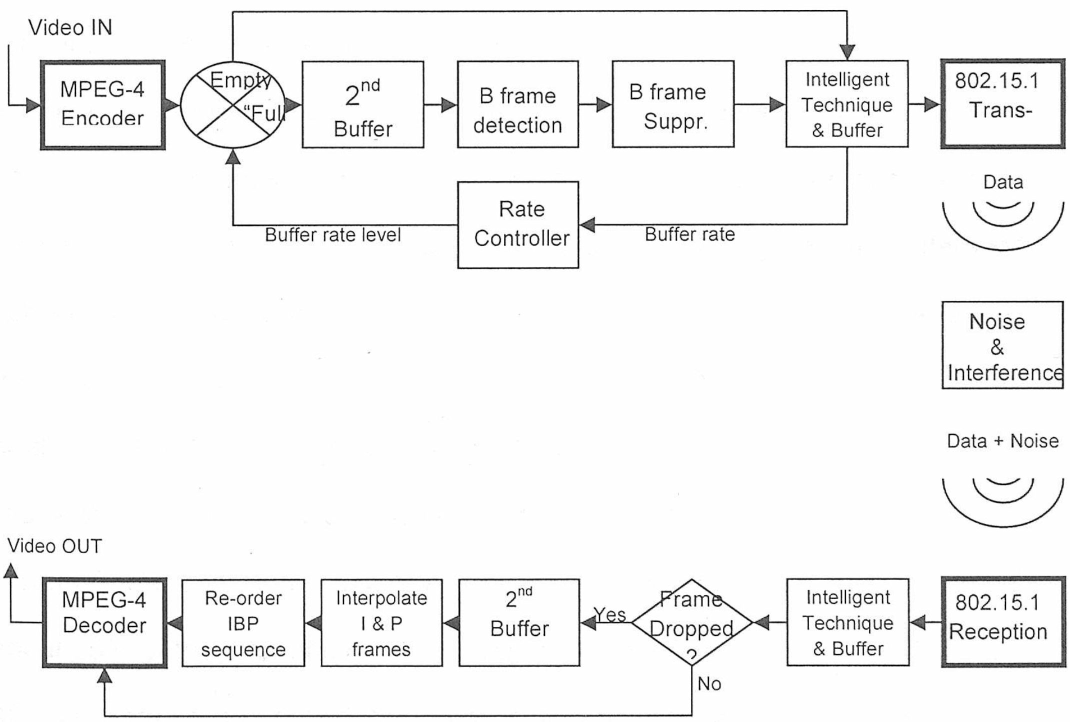

Figure 4.0: Block Diagram

In Figure 4.0, one has the block diagram of the complete system. This algorithm is decomposed in blocks. For each block, one have a specific function to take part of the system, except for four blocks (MPEG-4 Encoder, MPEG-4 Decoder, 802.15.1 Transmission and 802.15.1 Reception) that can be decomposed on sub-diagrams. Each sub-diagram will be explained in detail in separate subsections below.

The input of the system is a multimedia file, this file has to be an Audio Video Interleave format (AVI). The file goes into the MPEG-4 Encoder, this block analyses each frame, creates Group Of Picture (GOP), which is composed by a sequence of three frame types I frame, P frame and B frame, ordered as on Figure 4.1. I frames are called "Intra-frames" and are independent and need no information from other pictures. They contain all the information necessary to reconstruct the picture. P frames are called predicted and contain the difference between the current frame and a previous reference I or P frame. If the earlier reference frame is removed as a consequence of editing, the P frame cannot be decoded. P frames contain around 50% of the information of an I frame. And finally the B frames called Bidirectionnally predicted only use between the current frame and earlier and later I or P reference frames. B frames contain about 25% of the information of an I frame. And then puts it into the right order after each frame is encoded (this block is discussed later in this chapter).

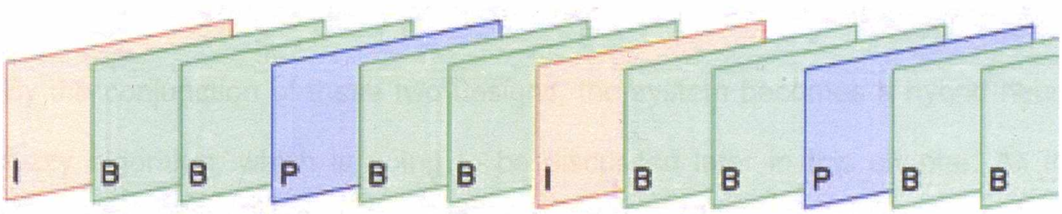

Figure 4.1: IBP frames sequence

Immediately after the encoding part, an evaluation of the different rate is made. This evaluation takes place here to determine the error level during the transmission. Indeed, if errors occur during the transmission, the level of the "token bucket" is low – the "token bucket" works like a bucket of tokens; more tokens you have in the bucket the more tokens you can take. If you have no token in the bucket, you obviously cannot take any token from the bucket.

An intelligent approach to rate control has been made by different artificial Intelligence systems. Some fuzzy logic controllers have been added firstly. After some tries with just rule-based fuzzy logic, the results, displayed and explained in the proper chapter, were not good enough, so some other artificial intelligences such as neural network techniques have been added. A neural network algorithm needs to be trained to learn of the evolution of the system. Here in the design, the decision to train 120 frames to get an accurate response of my method has been taken. So if the adaptation of the transmission rate is operating. For example if one has already data stored in buffers and then one has again too many data to be transmitted, then one adds more data in buffers until the neural network is pointing any free space in the bandwidth to send the extra data stored.

By the conjunction of these two designs, the system becomes a hybrid neural fuzzy algorithm, which is going to be discussed later in this chapter. All the results will be discussed in chapter 5, but what one can say at this stage is that the conjunction of the two intelligent techniques allows obtaining of better data during the training of the system.

In order to evaluate the rate correctly, an additional second buffer has been implemented before the token bucket already included in the hardware of the 802.15.1 standard. This second buffer, called here "added buffer", has been added in order to stock the data if the level of the token bucket becomes too high.

4.1 Encoder MPEG-4

Figure 4.2 is the simplest way to explain the MPEG-4 Codec I made done for this research.

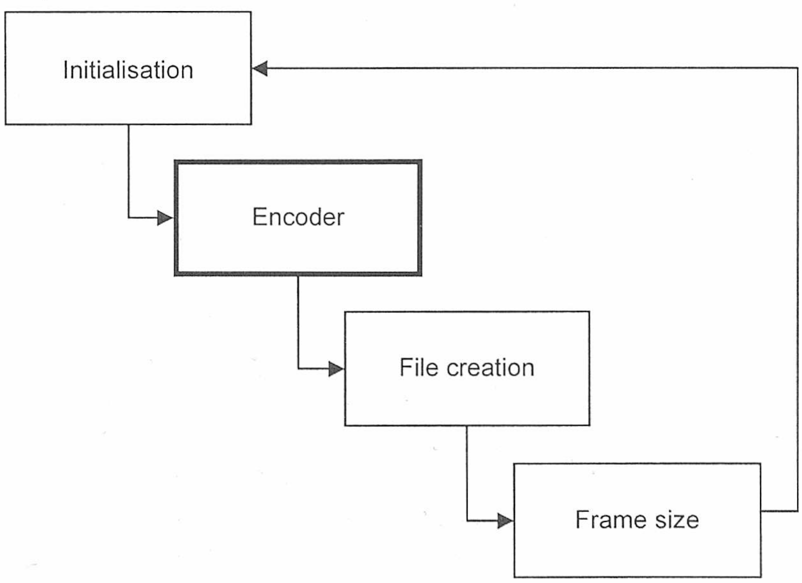

Figure 4.2: Block diagram of a MPEG-4 codec

In the above block, an initialisation is done. In the second box, is the Encoder, this block concentrates all the coding part. One is going to explain this part in depth on the next paragraph as it needs to be developed to be clearer. In the "file creation" block, the file is edited after the encoding. In the last part, the frame is given a size to correspond to the requirement of the user here. In this research the size is 240 pixels x 180 pixels.

Figure 4.3 shows the different steps to complete the encoding function. The first block resizes the frame in order to be treated. Then one is starting with the first frame to be encoded the I frame (I1). Then one has the second frame to be encoded the P frame (P4), followed by another I frame (I7). One then encodes the frame P10 and finally the I13 frame. Then one has the 8 B frames to be encoded. This order is due cause one needs to have the encoded I and P frames to encode the B frames.

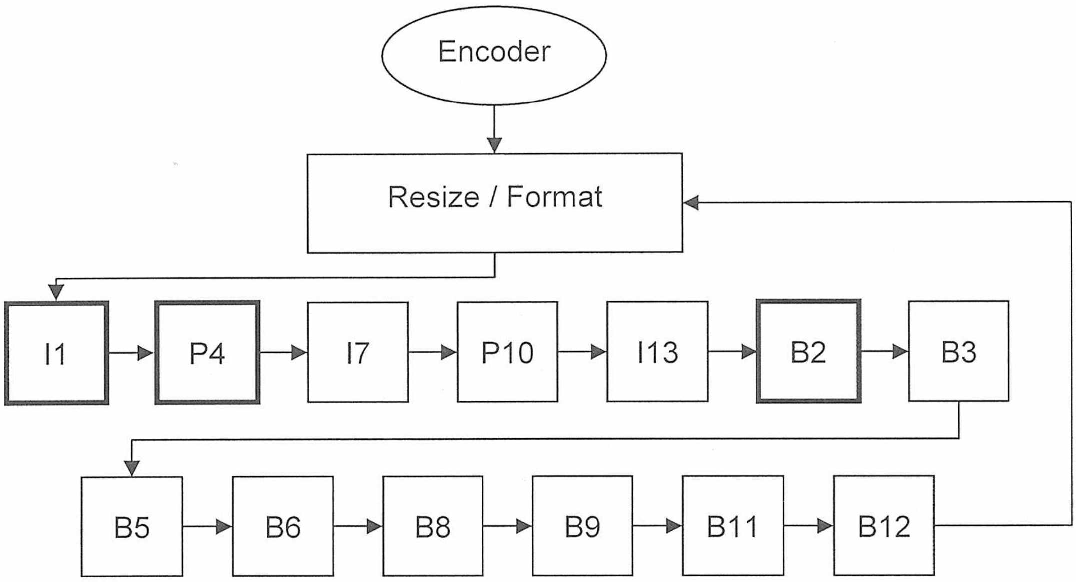

Figure 4.3: Block diagram of the encoder

- I Frame

Figure 4.4 explains how an I frame is coded.

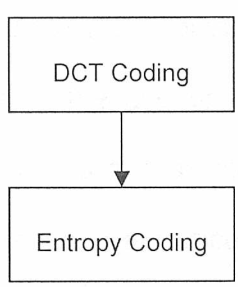

Figure 4.4: Block diagram of an I frame coding

- DCT Coding

DCT (Discrete Cosine Transform) encoding is probably the most well-known and the most intuitive coding method. When spatially encoding, the amplitude of a sample at a particular point in time or space is recorded, i.e., over time for audio waves and over space for images. As the samples build up over time or area, a digital approximation of the original waveform is produced. The frequency and accuracy of the samples obviously affects the quality of the approximation.

- Entropy coding:

This was developed by David A. Huffman for computer science and information theory. Huffman coding is an entropy-encoding algorithm used for lossless data compression. The term refers to the use of a variable-length code table for encoding a source symbol (such as a character in a file) where the variable-length code table has been derived in a particular way based on the estimated probability of occurrence for each possible value of the source symbol.

- P frame

Figure 4.5 explains how a P frame is coded.

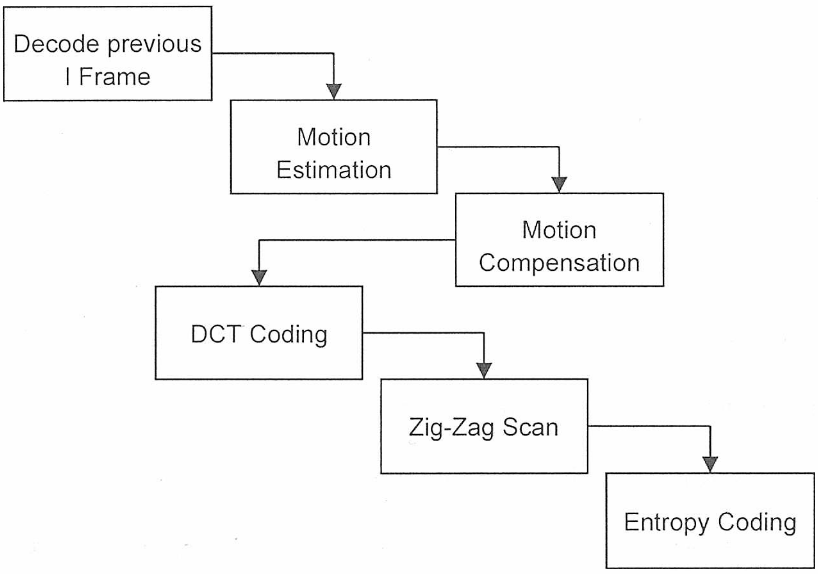

Figure 4.5: Block diagram of a P frame coding

• <u>Decode previous I frame:</u>

In order to decode the previous I frame, one simply needs to inverse the few steps done to encode the I frame. Here one has to follow the special decoding and the Huffman decoding explained in the decoding part (section 4.7) of this chapter.

• <u>Motion estimation:</u>

The temporal prediction technique used in MPEG video is based on motion estimation. The basic premise of motion estimation is that, in most cases, consecutive video frames will be similar except for changes induced by objects moving within the frames. In the trivial case of zero motion between frames, it is easy for the encoder to predict efficiently the current frame as a duplicate of the prediction frame. When this is done, the only information necessary to transmit to the decoder becomes the syntactic overhead necessary to reconstruct the picture from the original reference frame. When there is motion in the images, the situation is not as simple. The way that motion estimation goes about solving this problem is that a comprehensive two-dimensional spatial search is performed for each luminance macroblock. Motion estimation is not applied directly to chrominance in MPEG video, as it is assumed that the colour motion can be adequately represented with the same motion information as the luminance. It should be noted at this point that MPEG does not define how this search should be performed.

• <u>Motion Compensation</u>

For many frames in a film, the only difference between one frame and another is the result of either the camera moving or an object in the frame moving. In reference to a video file, this means that much of the information that represents one frame will be the same as the information used in the next frame. Motion compensation takes advantage of this to provide a way to create frames of a film from a reference frame. For example, in principle, if a film is shot at 25 frames per second, motion compensation would allow the file to store the full information for every I frame. The only information stored for the frames in between would be the information needed to transform the previous frame into the next frame. If a frame of information is 1 MB in size, then uncompressed, one second of this film would be 25 MB in size. Applying motion compensation, the file size for one second of the film can often be reduced to 8 MB, for typical video material. In MPEG, images are predicted from previous frames (P frames) or bidirectionally from previous and future frames (B frames). After predicting frames using motion compensation, the coder finds the error (residual) which is then compressed using the DCT and transmitted.

- DCT Coding

See the explanation above under "I Frame".

• Zig-Zag Scan:

The purpose of the Zig-zag Scan is to group low frequency coefficients in top of vector, and to map an 8 x 8 matrice to a 1 x 64 vector.

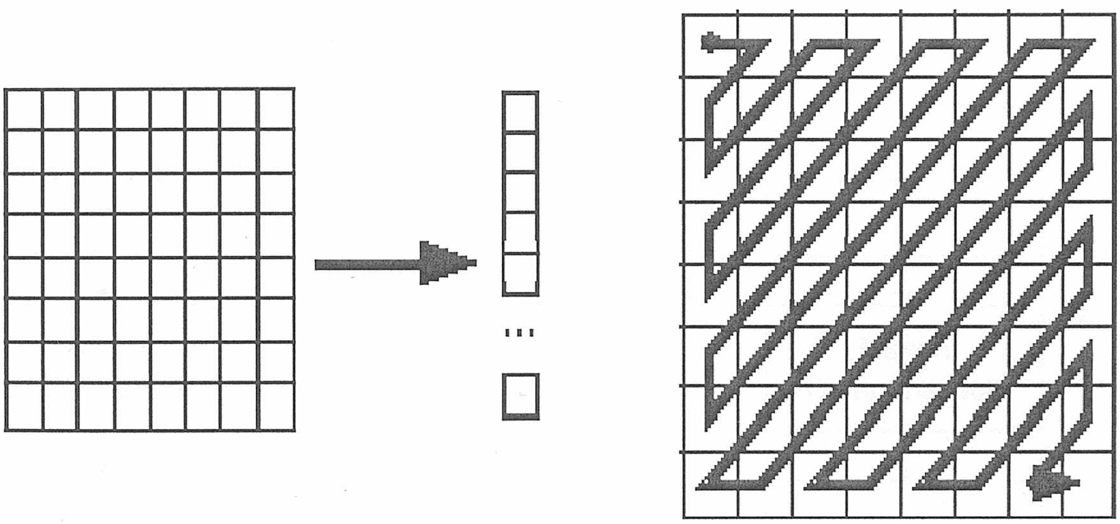

Figure 4.6: An example of a Zig-Zag scan

• Entropy Coding

See the explanation above under "I Frame".

- B frame

Figure 4.7 explains how a B frame is coded.

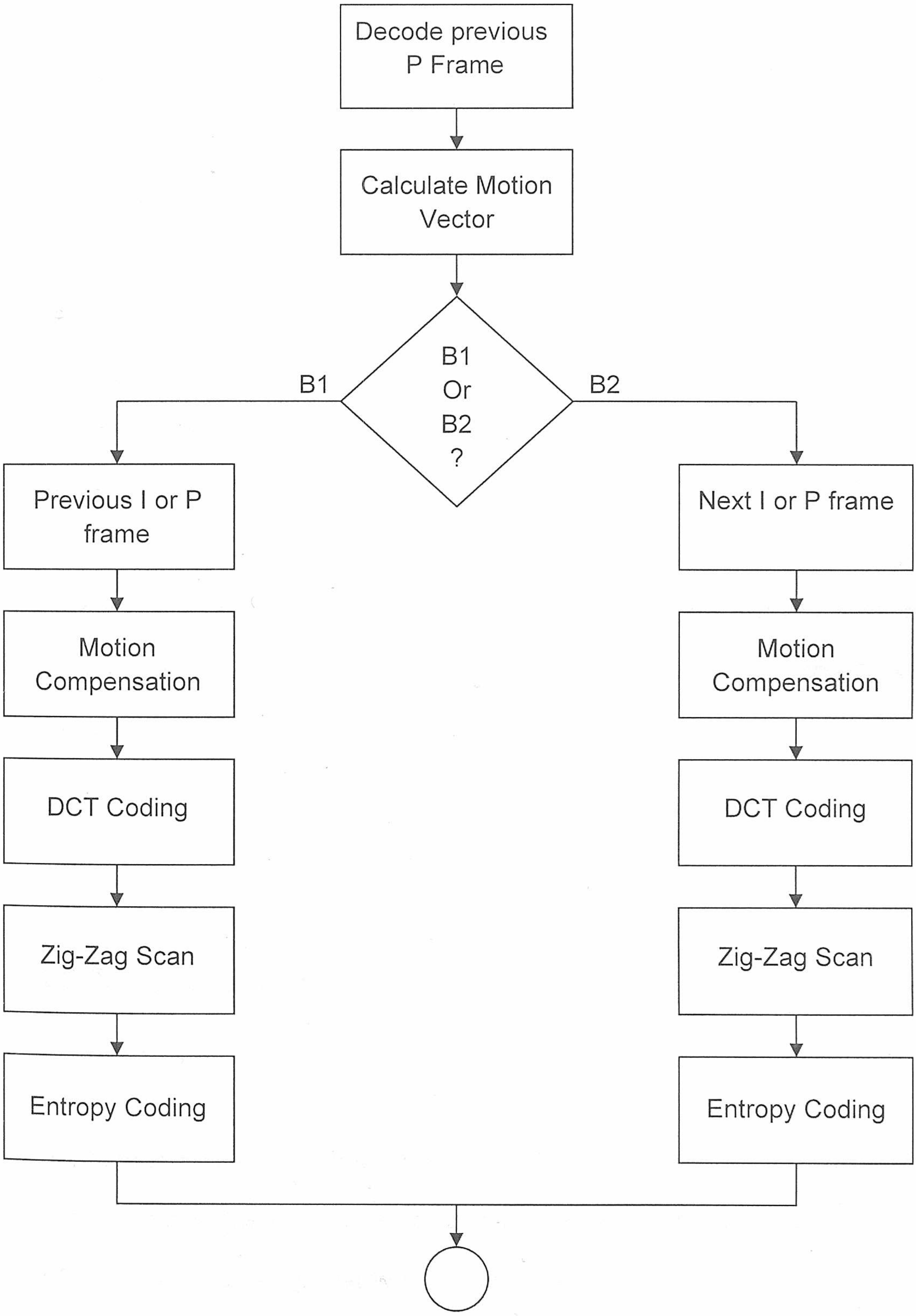

Figure 4.7: Block diagram of a B frame coding

For the B frame encoding, the algorithm is very similar to the P frame, except that one needs to decode the previous P frame. One needs to decode the P frame in order to obtain coefficients from the previous encoded frame. After the decoding part a calculation of the motion vector has been made, in order to quantify the difference between the previous or next frame and the B frame one wants to encode. When the motion vector coefficients have been calculated, one has to choose between the previous or the next I or P frame. If the B frame is the first in the group of two B frames, the Previous I or P frame will be used to encode. In opposite case, if the B frame is the second in the group, the next I or P frame will be used to encode the B frame. After the selection the classic algorithm is followed, motion compensation, DCT coding, zig-zag scan and entropy coding (each block of the algorithm has been explained in this chapter, please see above for more information).

4.2 "Added buffer"

The "added buffer" box shows the place of the buffer added in order to store and manipulate the data. This added buffer is used if the token bucket in the IEEE 802.15.1 is being charged. The management of the buffer-added load is done by neural network and fuzzy logic rules (these two will be developed deeply later on this chapter). This part of the algorithm will be explained in depth later in the chapter. The added buffer can contain only three groups of pictures. A buffer too large would create more delay in accessing the data, since one wish to reduce the delay the system could not have a big buffer. This box is not used for the simulation without Artificial Intelligence

4.3 B frame detection and suppression

This part of the algorithm is designed to reduce the size of the data sent to the token bucket if it is already full. As the B frames are interpolated between the I frames and the P frames, it could be possible to recreate the B frame at the reception end. This technique allows the token bucket to reduce the amount of data by reducing the arrival rate during a rush.

4.4 Intelligent technique and token bucket

In this part one will talk about how the design works by using a neural network and fuzzy logic to manage the token bucket and the "added buffer". As explained previously about the fuzzy logic rules, the algorithm will analyse precisely the level of membership of the token bucket and the "added buffer". The two figures below represent the four different states in which the buffer can be. For the “added buffer”, Figure 4.8, one has the first state called “Low” which means that the amount of data in the buffer is low so maximum space is available. In the second state, the amount of information stocked in the buffer is higher but not too much, called the “Medium” stage. The third state is when the space available in the “added buffer” becomes very small. The free space is less than thirty percent of the total of the buffer, it is called “High” because at this level the amount of information is really high and problems could occur if one has to stock more information inside the buffer. The last state is called “Very high” and it is, as one would imagine, due to the amount of information stocked in the “added buffer” being really high. That state is a critical state as, if more information has to be stocked in the buffer, an overflow would occur and data would be lost.

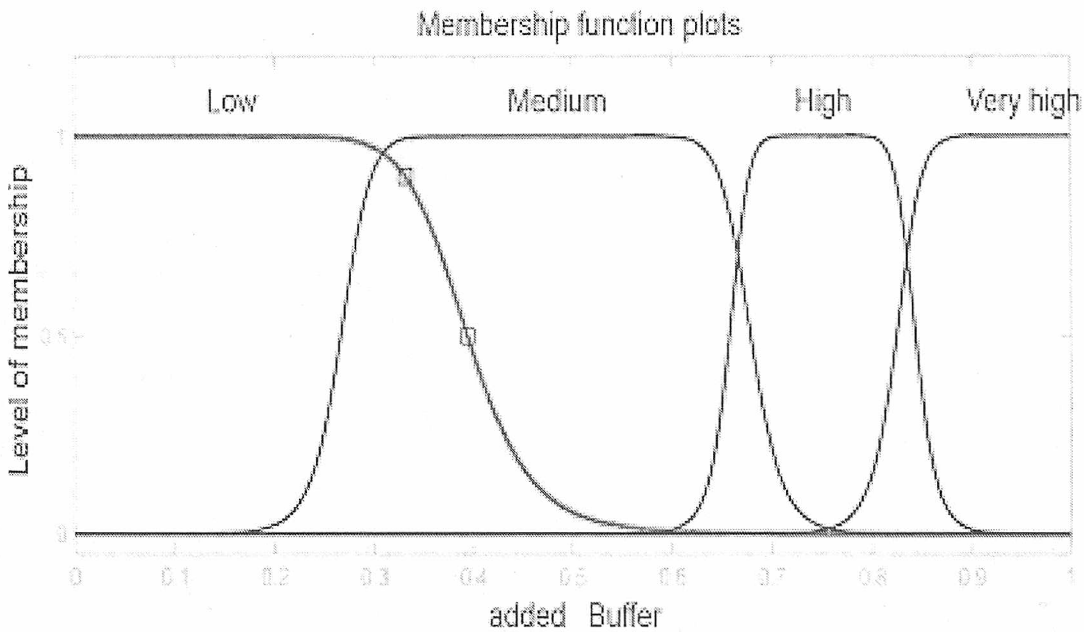

Figure 4.8: Level of membership of the “added buffer”

In Figure 4.9, one has approximately the same evolution as explained for the "added buffer". On the first part of the figure, one have the "Low" signal that represents when the token bucket is full of tokens and consequently there is plenty of space to store data before being transmitted. On the second part of the figure, the "Medium" signal shows that when the token bucket gets some data in its bucket, there is less room available to store data but it is not a critical level. The "High" level signal is more significant in terms of tokens available, as the amount of tokens left in the bucket is becoming lower than a third of the total bucket available. This level does not create data loss but if more data needs to be stored, one will have to be on the fourth signal. The fourth signal is called the "Very high" level, which means the amount of information stored in the token has reached the maximum available. If more information arrives to be stored, the token bucket will not be able to treat it and an overflow problem will occur.

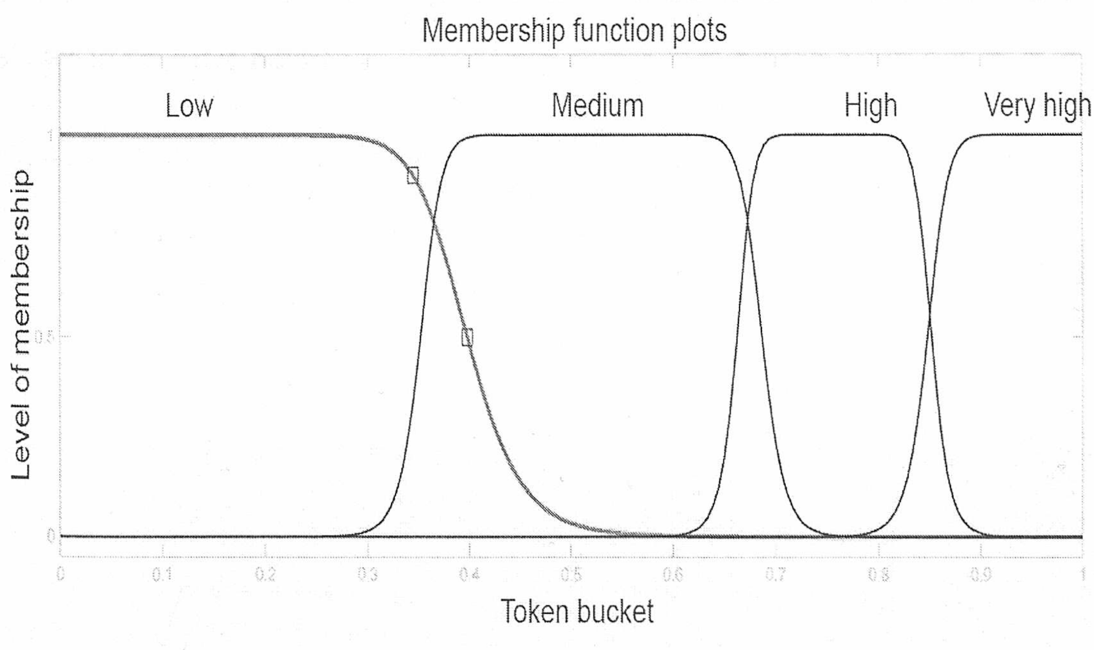

Figure 4.9: Level of membership of the "token bucket"

These two graphs (Figure 4.8 and Figure 4.9) can be summarised by a table (Table 4.1) representing this summary of the rules of the token bucket and the "added buffer" at inputs and the output of the fuzzy controller. One could easily see the evolution of the output following the data level in the token bucket and in the "added buffer". One can see that IF the token bucket is full of tokens (plenty of space available) AND the "added buffer" is empty (full space available) THEN the output of the controller is very fluid, that means the transmission is going smoothly without congestion or data loss. However, IF the token bucket is just a bit more than half full (more than half of it available for data) AND the "added buffer" has a high level of data stored THEN the output appears as rather congested, in other words, delay and data loss could occur in a short time if the transmission stays in the same condition with the same amount of data arriving from the encoder. Big congestion resulting in effective data loss and degradation of the quality of picture occurs IF the token bucket is empty (no token left in the bucket) AND the "added buffer" is full of data, as is shown in the table below.

Rule Number		Token Bucket		Buffer		Output
1		Low		Low		Very Fluid
2		Low		Medium		Intermediate
3		Low		High		Fluid
4		Low		Very High		Intermediate
5		Medium		Low		Intermediate
6		Medium		Medium		Small Congestion
7		Medium		High		Small Congestion
8	IF	Medium	AND	Very High	THEN	Congestion
9		High		Low		Fluid
10		High		Medium		Small Congestion
11		High		High		Congestion
12		High		Very High		Congestion
13		Very High		Low		Intermediate
14		Very High		Medium		Congestion
15		Very High		High		Congestion
16		Very High		Very High		Big Congestion

Table 4.1: "IF-AND-THEN" rules

At this stage, one has a picture of the degree of starvation of the buffers which is very important information to optimise the bandwidth used during the transmission. As now one have an updated picture of transmission state, one can regulate the flow of information sent by the device following the bandwidth requirement. The bandwidth usage varies constantly so an adaptive method must be deployed to follow as good as possible the real time transmission rate. That is the moment one introduced the neural network. The N-N here will follow instantly the requirement of the transmission system by using the buffers level and allow more or less data to go through the device to be sent.

As said earlier in the chapter a neural network algorithm must be trained to get the best results at the output. Here one trained the data the buffers' rate from 10 GOP, which means around 120 frames. Every 120 frames a new set of data has been collected by the system to be trained again. The system is then training data get from the previous encoded GOP. Then after the training the system is waiting for the next GOP encoded, and so on until the tenth GOP.

4.5 IEEE 802.15.1 Transmitter

Figure 4.10 is the diagram block representation of our transmitter algorithm.

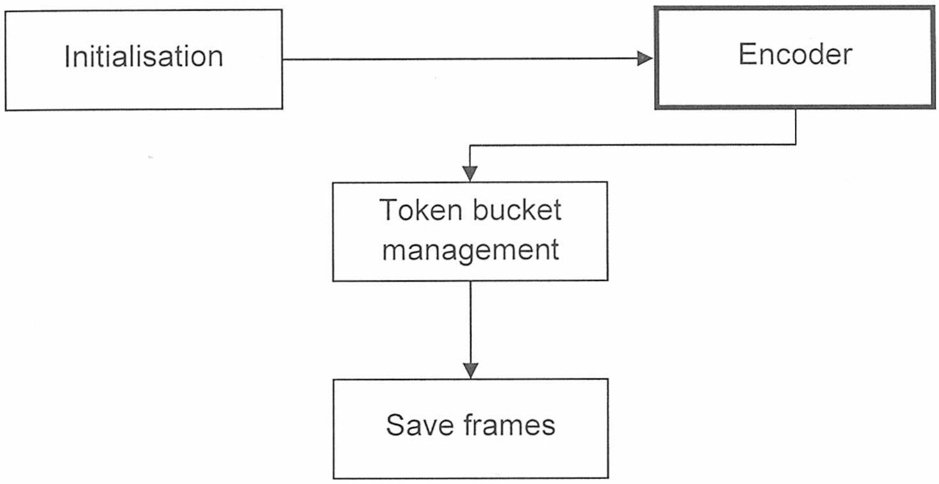

Figure 4.10: Diagram block of the IEEE 802.15.1 Transmitter

- **Initialisation:**

This diagram block explains the way this project is conducted to simulate the IEEE 802.15.1 standard transmission. In the first step, one obviously need to initialise few parameters, all those parameters are entered at the beginning of the simulation. The initialisation will be the bandwidth maximum value available in the IEEE 802.15.1 standard, here one selected 724Kbps which is the download value of the asynchronous mode. The second parameter is the first group of picture, this parameter let the simulation start at a different point of the video clip. The next parameter is obviously the last group of picture, this parameter lets the simulation end at a different value. These two previous parameters offer the possibility of testing our system at different parts of the video clips and thus to simulate different action in the video (lots of action or not much action). One needs, at the initialisation point, to load the quantisation parameters for the Variable Bit Rate (VBR) transmission, the value of Qi, Qp and Qb are loaded during this part of the program. This initialisation is only used for the VBR transmission (non-AI system), because in this adaptive design these parameters are variable and depend on the actual transmission parameters.

- **Encoder:**

The Encoder has been described above.

- **Token bucket management:**

The management of the token bucket is done by a comparison of the rate of the data arriving at the entrance of the bucket and the maximum bandwidth available. According to this comparison, data is sent or put in the token bucket and queues to be sent by the device.

• **Save frames:**

This part gives a common name to all the frames to enable the receiver to recognise and decode each frame in the right order.

4.6 IEEE 802.15.1 Rx

Figure 4.11 is the diagram block representation of our transmitter algorithm.

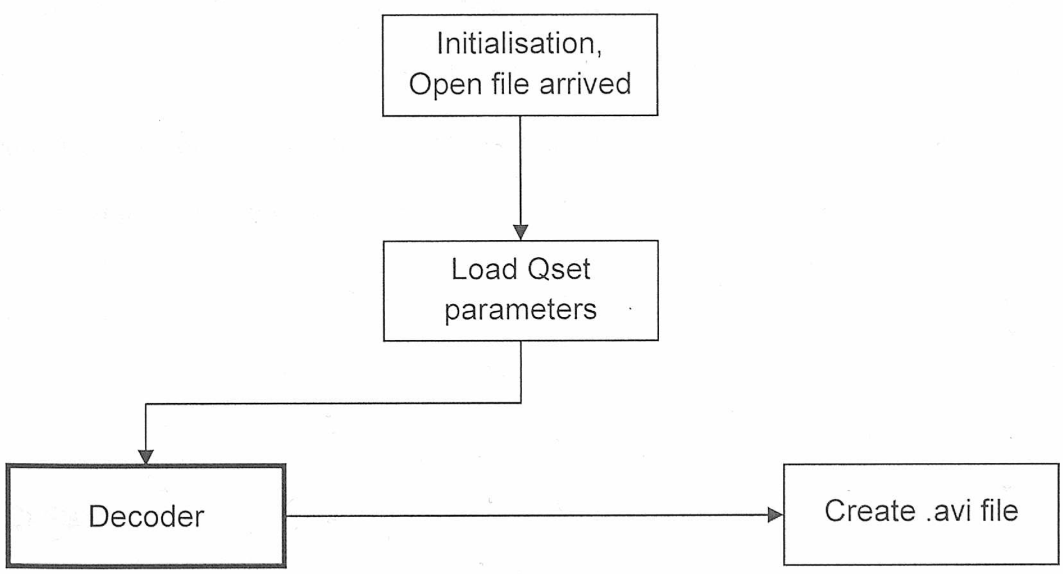

Figure 4.11: Diagram block of the IEEE 802.15.1 Receiver

- **Initialisation, open file:**

This part of the program is to initialise the receiving part of the program. At this level, the user is asked to enter the name of the file received. According to the sender part, the user will have to enter the first group of picture and the last group of picture of the video clips named previously.

- **Load Qset: parameters**

At this stage, the value of the quantisation parameters is asked for the Variable Bit Rate (VBR) mode. The user is asked to enter the Qi, Qp and Qb parameters in order to decode properly the clip sent. Quantisation parameters are not requested on our intelligent system as the parameters depend on the traffic condition during the transmission.

- **Decoder**

The Decoder will be described below (section 4.7).

• **Create .avi file**

This part of the program gathers all the frames together (already in the right order, see under "Decoder") from the first group of picture to the last group of picture asked by the user. The program then writes the total clip on an AVI file format. This part of the transformation is made directly from a MATLAB function.

4.7 Decoder MPEG-4

Figure 4.12 is the diagram block representation of the decoder algorithm.

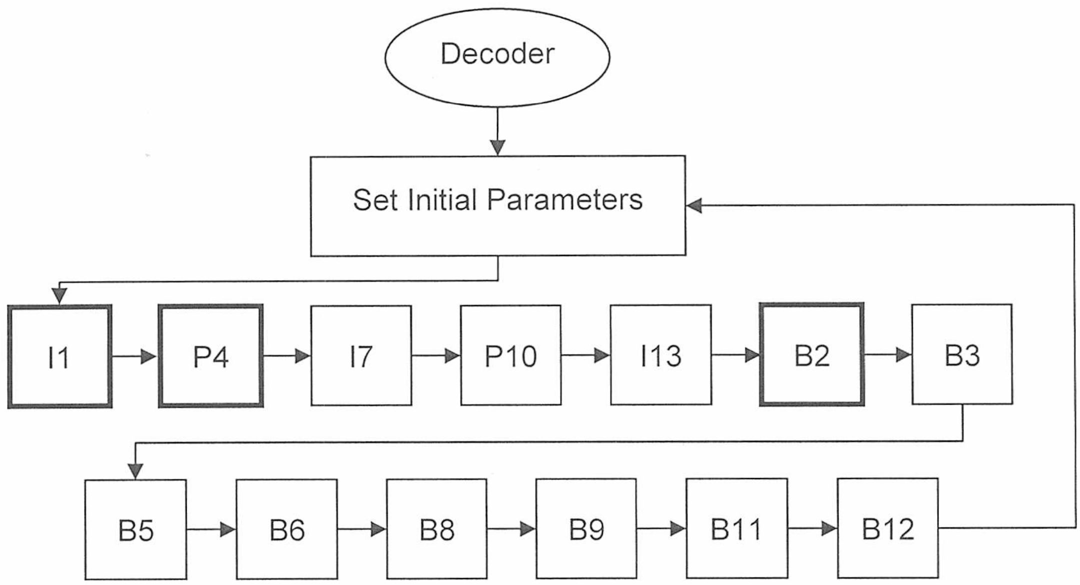

Figure 4.12: Diagram block of decoder

- **I Frame**

Figure 4.13 explains how an I frame is decoded.

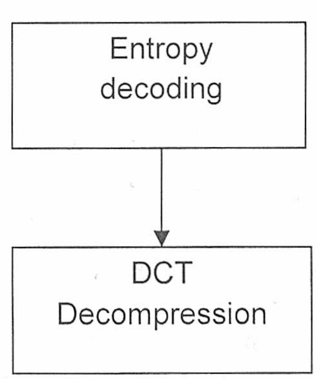

Figure 4.13: Diagram block of the I frame decoding

• Entropy decoding

This entropy decompression is to decompress the picture using the coefficients obtained in the DCT decompression. This decoding follows the entropy decompression scheme explained below in this chapter.

• DCT Decompression

DCT decompression is the Inverse of the DCT coding shown previously. This part will test if one is decoding an I frame or a P or B frame. If one is trying to decompress a B or P frame, one is getting the DC coefficient and the Motion Vector (MV). If one is decompressing an I frame, one does not need to get the Motion Vector as the I frame is a reference. So, in the case of an I frame, one only gets the DC coefficient.

- P frame

Figure 4.14 explains how a P frame is decoded.

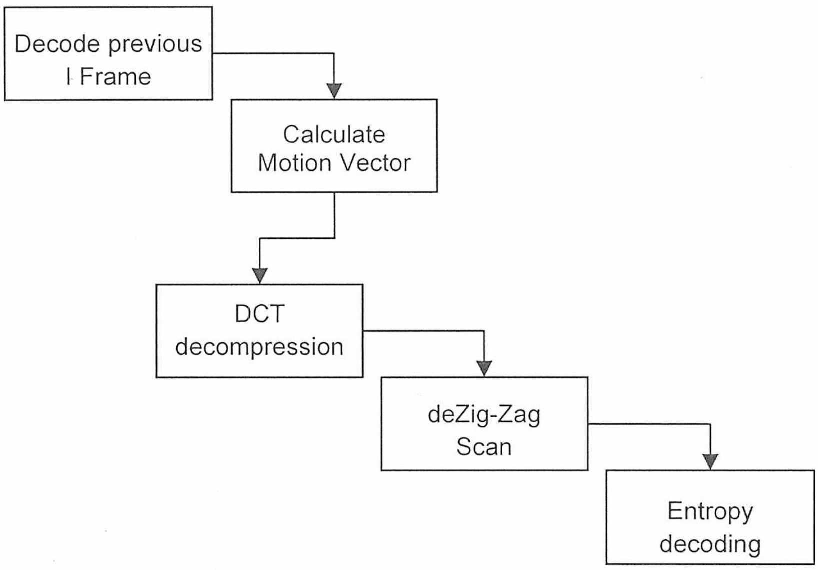

Figure 4.14: Diagram block of the P frame decoding

- <u>Decode previous I frame:</u>

In order to decode the previous I frame, one simply needs to follow the steps already done in the previous explanation by doing the Huffman decoding and the DCT decompression.

- Motion Vector:

As one is decoding a P frame, one needs to get the Motion Vector (MV). MV has been explained previously in chapter 2. This coefficient analyses the difference between the I frame and the P frame one are decoding.

- DCT Decompression

See the explanation (section 4.7: I frame).

- deZig-Zag Scan:

The purpose of the deZig-zag Scan is to inverse the Quantisation and inverse the Discrete Cosine Transform (DCT).

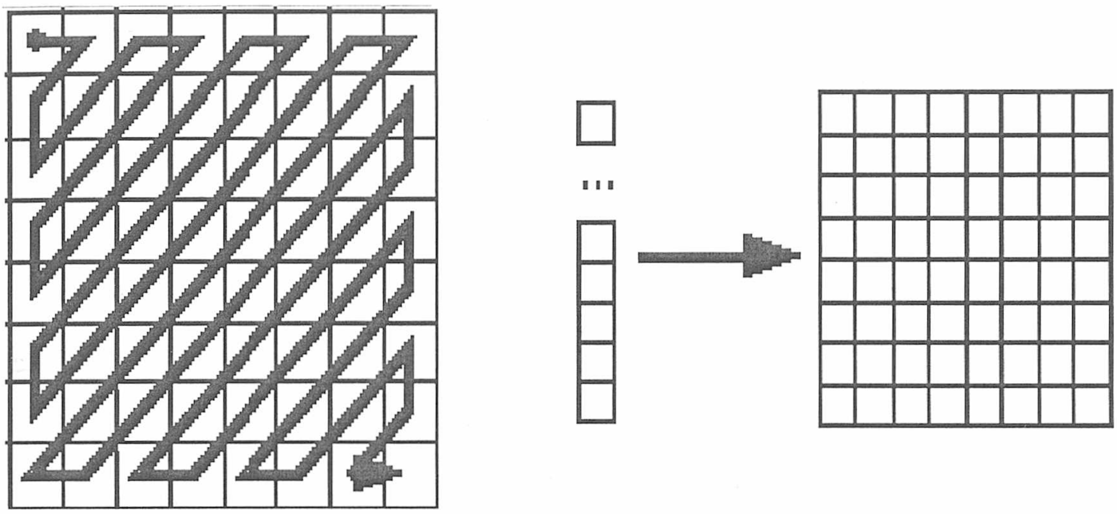

Figure 4.15: Diagram block of a deZig-Zag scan

• Entropy decoding

See the explanation (section 4.7: I frame).

- B frame

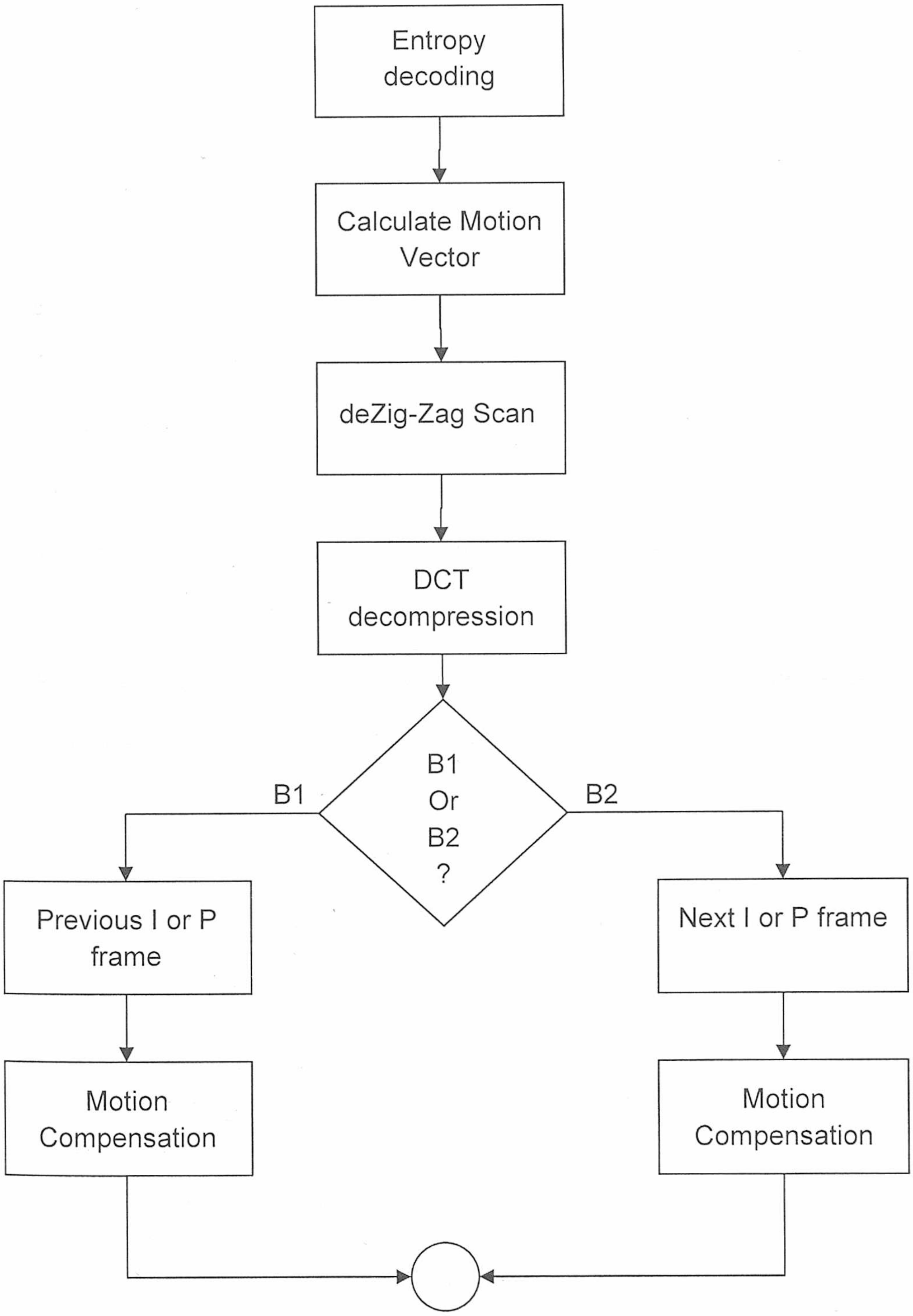

Figure 4.16: Diagram block of the B frame decoding

For the B frame encoding, the algorithm is very similar to the P frame, except that one needs to decode the previous or the next P or I frame. One needs to decode the P or I frame in order to obtain coefficients from the previous encoded frame.

After the entropy decoding part, the classic algorithm is followed, calculates the motion vector, dezig-zag scan, DCT decompression (each block of the algorithm has been explained in this chapter, please see above for more information). After the DCT decompression, one needs to choose between the previous or the next frame to realise the temporal decoding. The selection is the same as the encoding part. When the previous or next frame is decoded, one uses the motion vector coefficient to compensate the image.

CHAPTER 5:

Computer simulation, Results

5.1 Voice transmission over IEEE 802.15.1 standard

The design used for this part has been based on a Matlab newsletter, edited by Stuart McGaritty in 2001 [82]

The maximum bandwidth of the IEEE 802.15.1 standard is 1Mbps. This standard transmits over radio channel voice and data information. IEEE 802.15.1 can transmit at as low as 1mW in a radius of 10 meters. Gaussian Frequency Shift Keying (GFSK) is the modulation method used in IEEE 802.15.1. In ISM band other devices can interfere during a transmission, frequency hopping is used to eliminate that problem. With different system of coding, if a problem of interference occurs, it can be accepted or a recovery can be achieved. Slots of 625µs are the time base for the hop frequency and it is as well dividing the transmission time. As said, the data transmission is up to 1Mbps, but the frequency hooping is using a 79MHz bandwidth in order to generate the effects. In this simulation one will be focus on the physical layer simulated by Matlab/Simulink. one will as well simulate the logical link control, link manager protocol in order to transmit an audio file.

A device using the IEEE 802.15.1 protocol can be indifferently slave or master when a communication occurs. In order to create the communication with one or few slaves, the master first creates the connection. Figure 5.1 shows the path between a transmitter (master) and the receiver (slave) to make a communication.

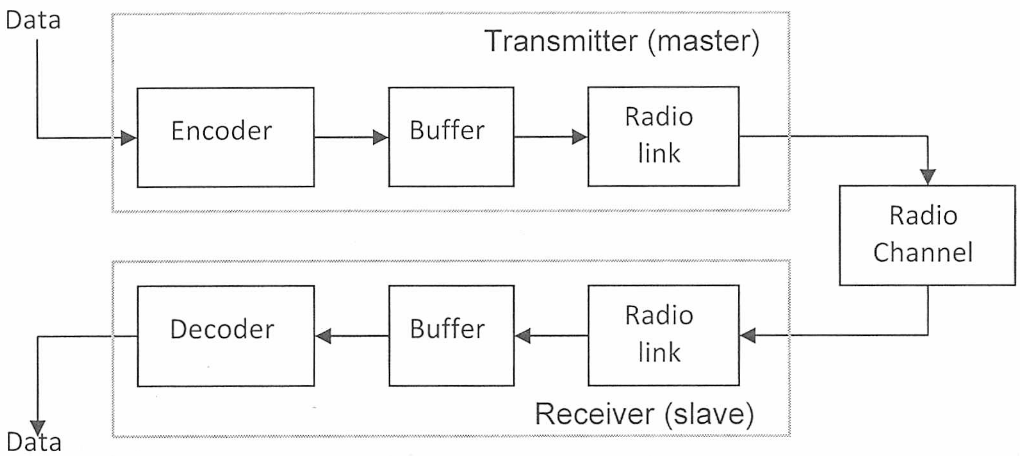

Figure 5.1: Communications link between master and slave.

Figure 5.2 shows deeply the structure of the transmitter through block diagram, Continuous Variable Slope Decoding (CVSD), buffering, noise clearing, concatenation, Header Error Check (HEC), Forward Error Correction (FEC), modulation & frequency hopping, and Radio Frequency (RF) subsystem.

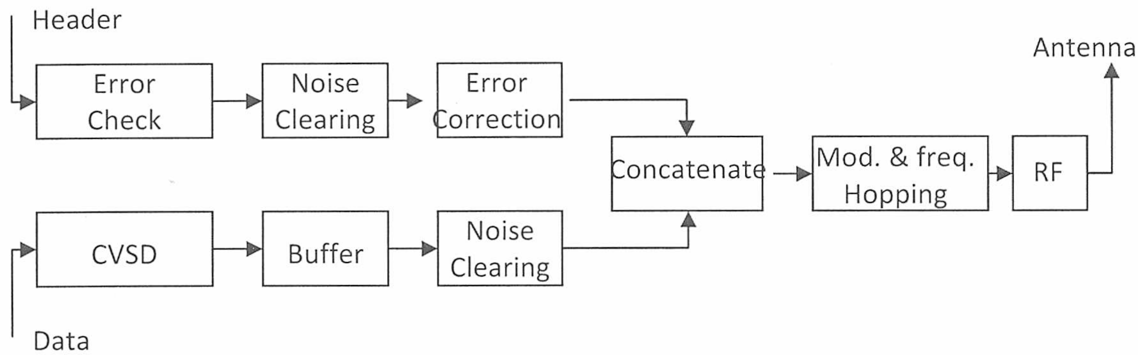

Figure 5.2: Transmitter specification

In IEEE 802.15.1, a Synchronous Connection Oriented (SCO) is a well known type of voice communication. The each SCO is transmitted every six slots. The period between 2 similar SCO is then 3.75ms (6x625µs), called TSCO (Time of Synchronous Connection Oriented). The return from the slave to the master is transmitted on the next slot, as shown in Figure 5.3. This technique can work with up 3 simultaneous communications. On the following figure, one can see the TSCO composed by the 6 slots, the 3 masters' slots (M1, M2 and M3) and the 3 slaves' slots (S1, S2 and S3).

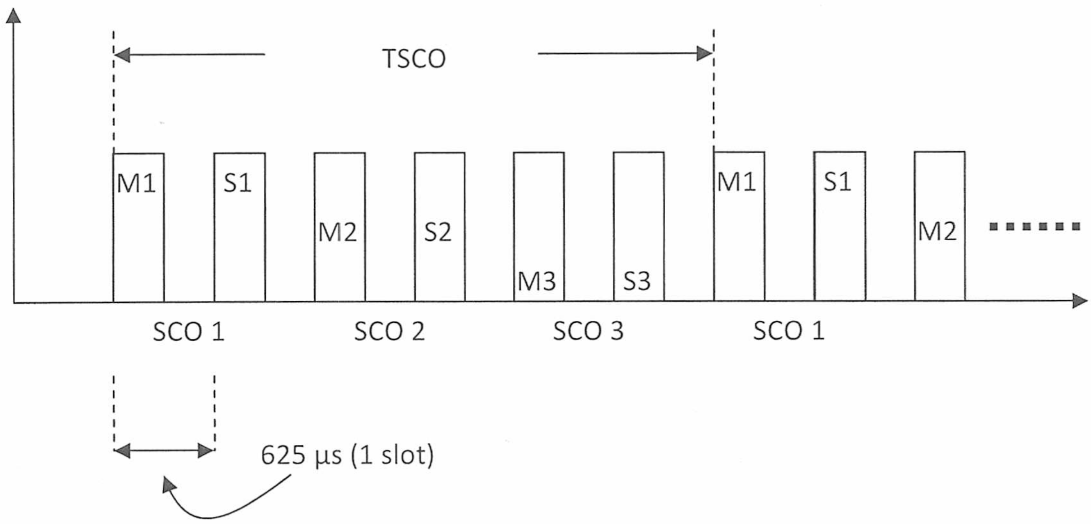

Figure 5.3: Timing diagram of three simultaneous voice calls.

In order to avoid problem during the construction of this design, the beginning of the design is the radio channel components and the external blocks. Then one adds the other components like the hopping frequency and the HEC. Each time a new module is completed a test has been made. Figure 5.4 shows the complete model. This model comprises the transmitter, radio channel, a generator of Gaussian noise and a simulator of 802.11b device communication, slave receiver, an instrumentation block and an error meters block.

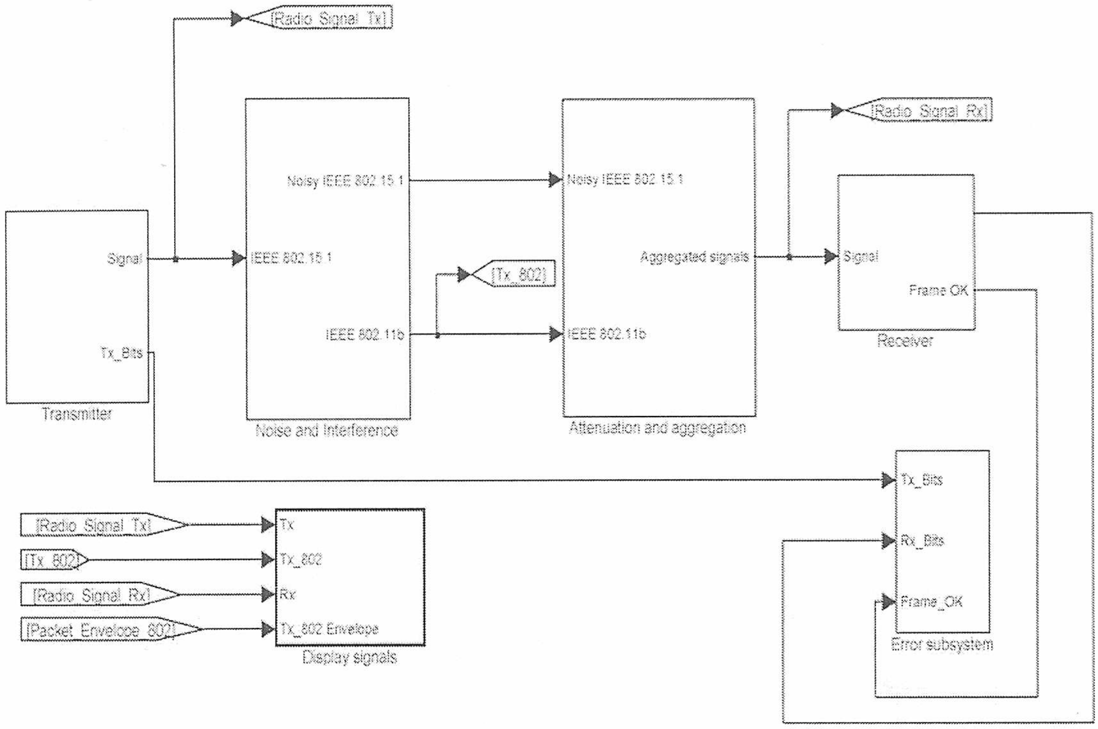

Figure 5.4: Main diagram for IEEE 802.15.1 transmission.

To be the realistic one added to the design an IEEE 802.11b generator who can be configured to be on and off. An attenuation to simulate the signal lost during the transmission has been added here an attenuation of 0,01W between the transmitter and the receiver, this value is represented by a gain of -20dB. To be more accurate, as well one added some Gaussian noise to simulate the global noise the channel can be affect by. This Gaussian noise can be parametered by it ration input / noise value (Es/No).

On the Figure 5.5, the function of the transmitter is simulated to follow the specification explained earlier in this section (cf. Figure 5.2). One could find for example the modulation and frequency hopping blocks as well as the formatting subsystem of the signal.

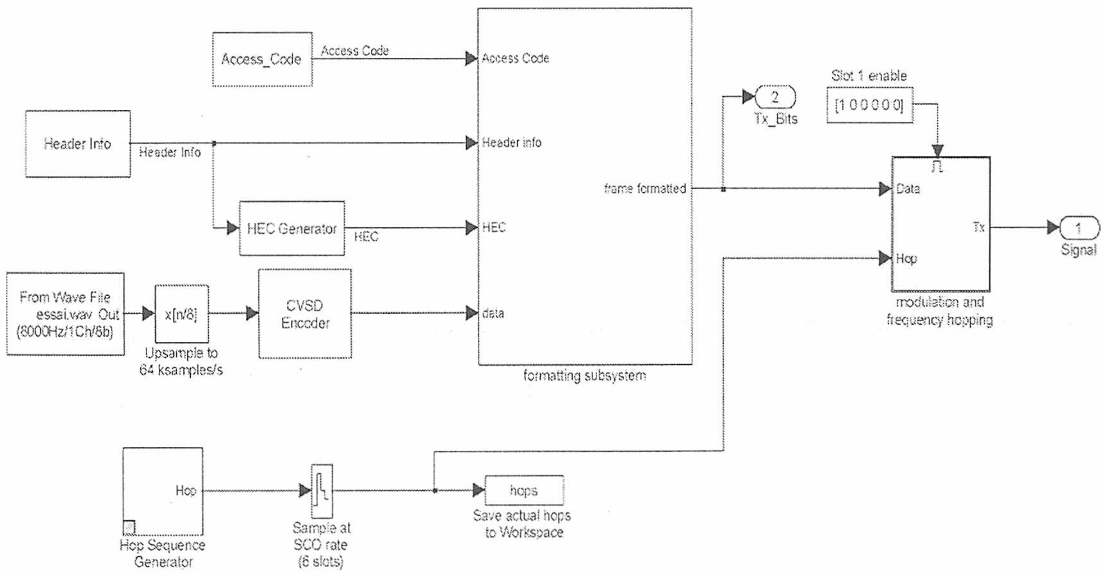

Figure 5.5: Master subsystem.

At the input of the transmitter, one has a signal of 8 KHz. As the coding speed is 64 KHz an interpolation of the input signal is required. By using a differential coding scheme, the coder encodes each speech sample with a single bit. Then if the sample increases a value of 1 will be transmitted and a value of 0 if it is decrease. This design is very accurate to minimise the impact of an error during the transmission. If at the reception end a bit has been corrupted, the amplitude range due to the corruption will be reduced by the design itself.

In order to follow the specification of the IEEE 802.15.1 standard, after being encoder the data are shaped into frame of 240bits at the frequency of 64KHz. These parameters correspond at the TSCO (3,75ms slot time). To facilitate the modification of the continuous variable slope decoding, a block containing all the modifiable parameters was created.

The 240 bits of data to be transmitted has to be formatted with additional control information as shown in Figure 5.6.

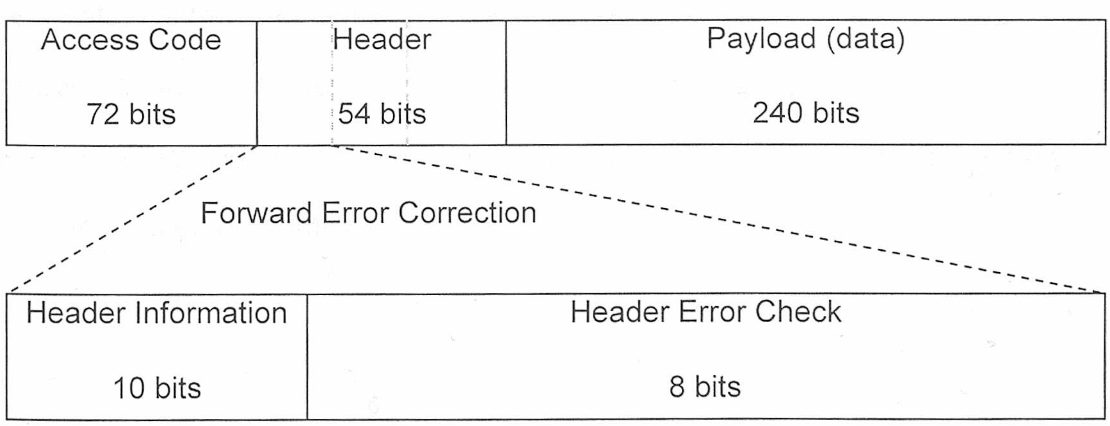

Figure 5.6: Access code, header, and payload framing specification.

As one can see on the Figure 5.6, a frame is 366 bits long composed by 72 bits of access code, 54 bits of header and 240 bits data wanted to be transmitted (payload). Inside the header, one could find the Forward Error Correction which is composed by 10 bits devoted to the header information. These 10 bits are composed by 3 bits for the slave address, 4 bits for the packet type and 3 bits for status. The header information is used to generate the 8 bits of the Header Error Check (HEC). This FEC is then replicate 2 more time to obtain the 54 bits required (18+18+18=54bits).

The whole frame is modulated using Gaussian Frequency Shift Keying (GFSK) and transmitted at 1 Mbps. The Figure 5.7 is showing the design of the modulation and frequency hopping used. Here the GFSK modulator generates a signal relative to the carrier of 150 KHz for a bit with a value of 1. On the opposite, the signal will be – 150 KHz if the value of the bit is 0. As the specification of the standard is to have slots of 625µs, frames of 62500 samples has been created. In order to obtain the correct specification, one specified 100 samples per symbol as the GFSK modulator generates a bandwidth centred on 0Hz with a range of -50MHz / +50MHz.

On Figure 5.7, one has another part which is the simulation of the hopping frequency. The hopping frequency is based on the idea that one can transmit information on different carriers at each pulse. This technique makes it almost impossible for another device from the ISM band to interfere during the communication. To generate this hopping frequency, one is mixing the signal with one of the 79 carriers. The carrier's frequencies are between -39MHz and +39MHz. If the input value of the hop is 0 then the frequency of the sine wave generated will be -39MHz. If the value is 1 then the frequency would be -38MHz, etc... To obtain the 79 symbols needed to generate all the possible carriers a Multiple Frequency Shift Keying (MFSK) modulation was used. This modulation will generate 79 values separated by 1MHz.

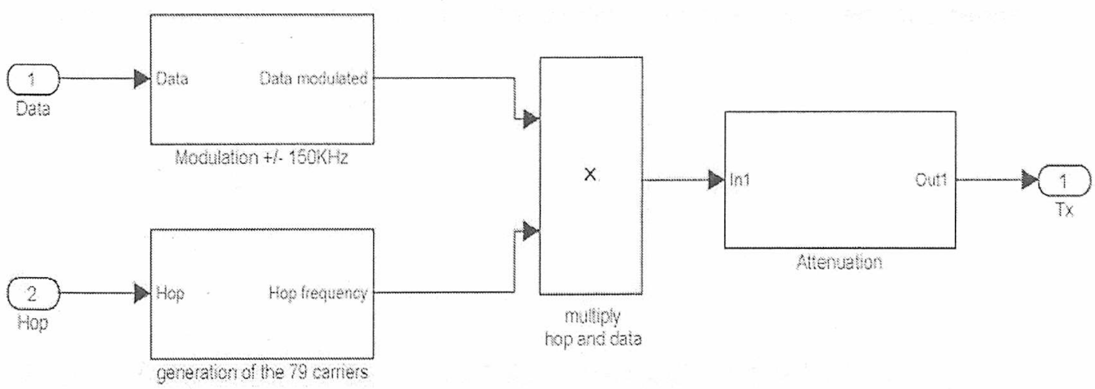

Figure 5.7: GFSK modulation and frequency hopping.

As said, the transmitter produces some pulse when he is ready. This moment is easily seen on the Figure 5.8 which shows the timing diagram of the IEEE 802.15.1 with no interference.

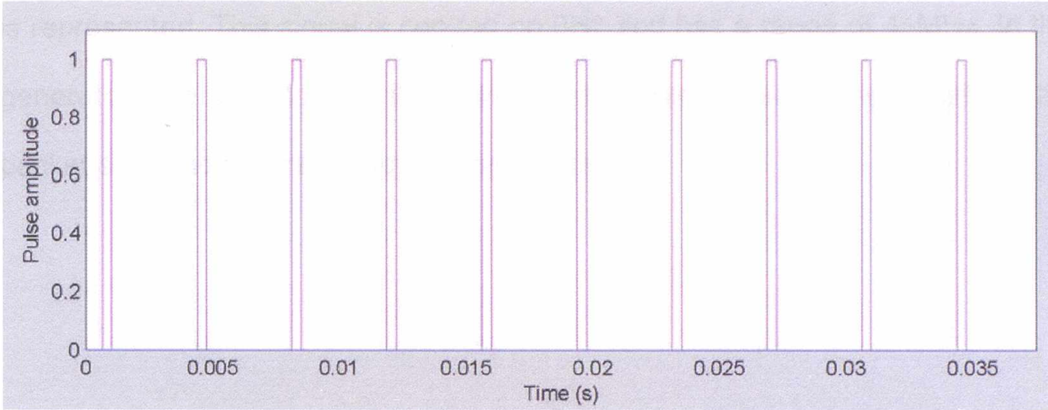

Figure 5.8: IEEE 802.15.1 signal

The next figure (Figure 5.9) is the representation of a hop frequency generated. One can see as explained before the bandwidth is centred on 0 and the range is +/- 39MHz.

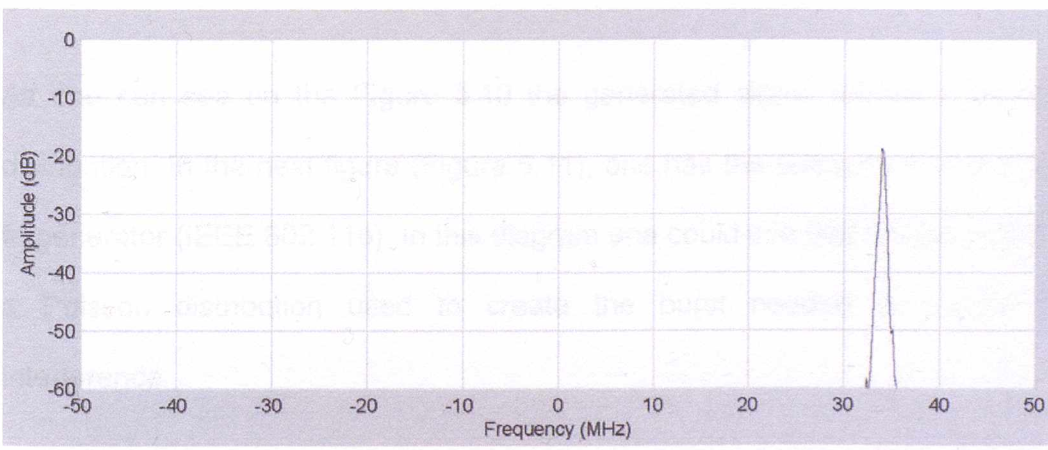

Figure 5.9: Hop frequency diagram

On figure 5.10, a model of the signal generated by the IEEE 802.11b generator is represented. This signal is centred on 0Hz and has a range of 44MHz. In this generator one can specify the power of the transmission, the length of the packet sent and the frequency on the channel.

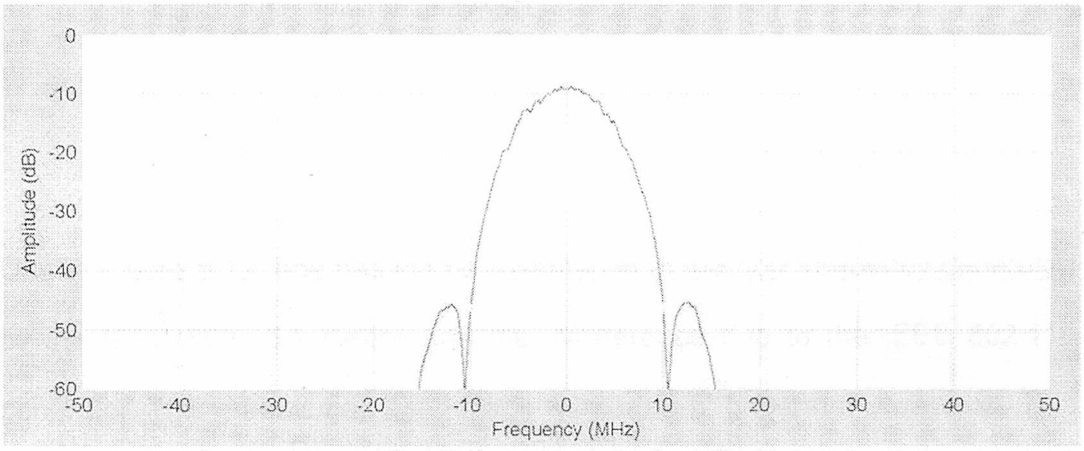

Figure 5.10: IEEE 802.11b frequency diagram.

As one can see on the Figure 5.10 the generated signal follows a Poisson distribution. In the next figure (Figure 5.11), one has the subsystem of our "Wi-fi" generator (IEEE 802.11b). In this diagram one could see that the generator is a Poisson distribution used to create the burst needed to create the interference.

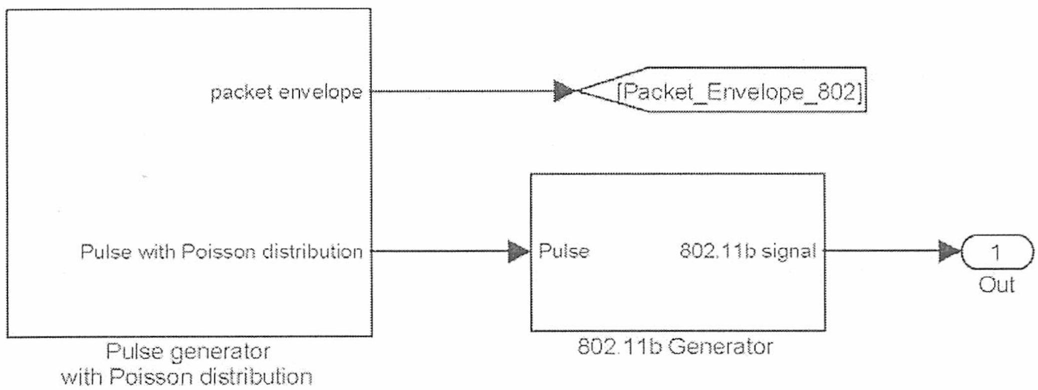

Figure 5.11: 802.11b generator following a Poisson distribution

On the Figure 5.12, one has the representation of the hop frequency generated by the IEEE 802.15.1 device and the interference due to the IEEE 802.11b generator. This graphic has been made with a "Wi-fi" generator always on. This case is highly improbable but it is really helpful to figure out the worth case.

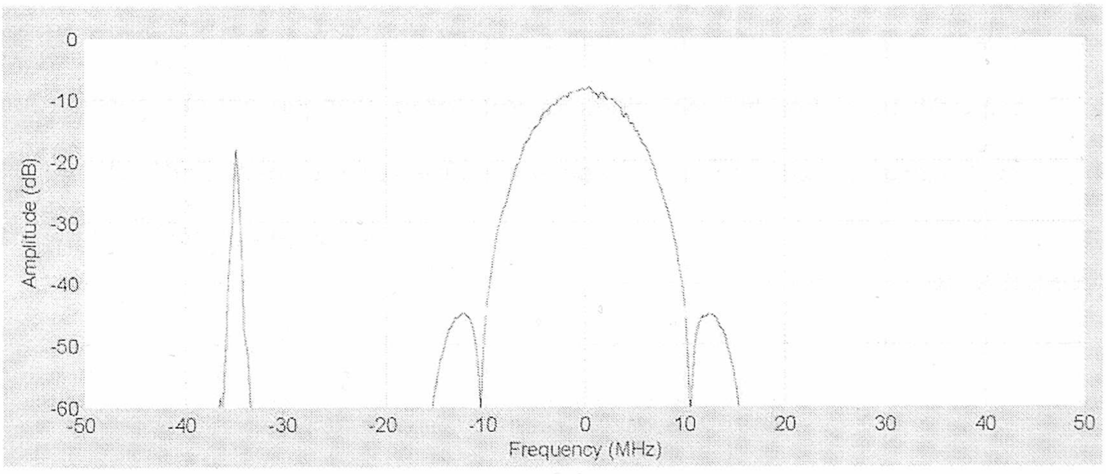

Figure 5.12: IEEE 802.15.1 and 802.11b spectrum plot.

On the Figure 5.13, the IEEE 802.15.1 device generates regularly the pulse of transmission and then one has the IEEE 802.11b who generates heretically its own pulse to create the inference. As one can see here, the "Wi-Fi" is not always on which is a more realistic response than the previous Figure 5.12.

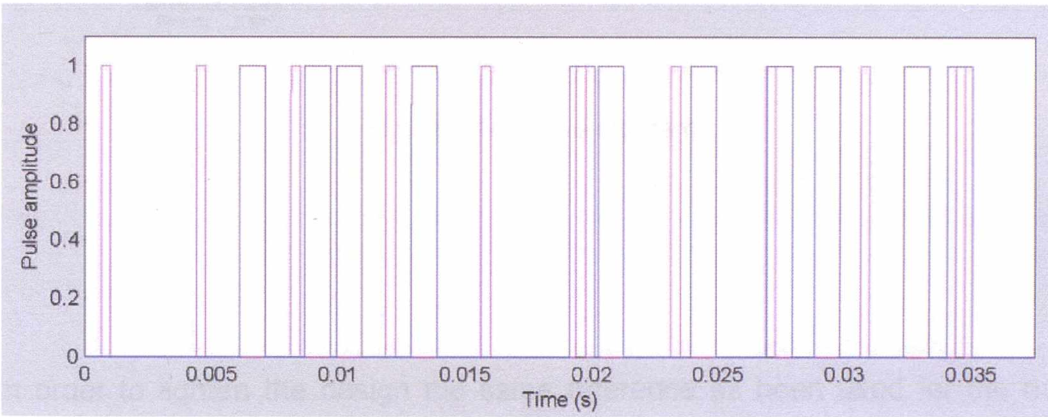

Figure 5.13: IEEE 802.15.1 and 802.11b timing diagram.

At the receiving end, the design follows a similar process than in the master, but obviously in the opposite order. For example, one demodulated then one deconcatenated, then one separated the header to the data transmitted, as shown on the Figure 5.14.

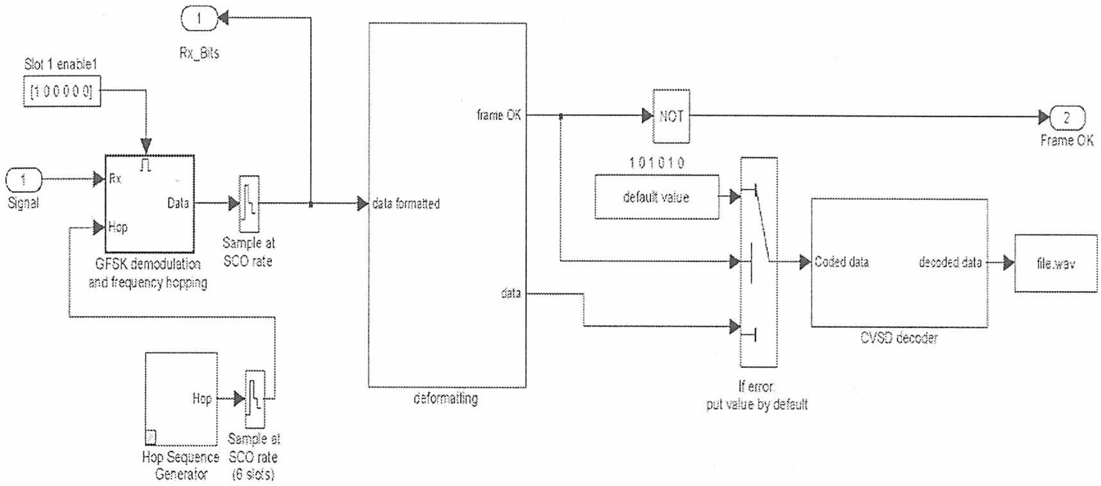

Figure 5.14: Slave diagram.

In order to lighten the design the same reference as been used for the hop frequency. If during the reception, the errors are too much to correctly demodulate, the header error check will be different to the header information. If it is the case, the data received will be ignored and a value will be applied by default.

5.2 Video transmission over IEEE 802.15.1 standard without intelligent technique

This part will explain the results obtained following the simulation scheme of this thesis without intelligent technique. In this part, one will explain the way the Variable Bit Rate (VBR) video transmission was built up. VBR is related to the bit rate used in video encoding. VBR files vary the amount of output data per sample. VBR allows a higher bit rate to be allocated to the more complex segments of media files while less space is allocated to less complex segments.

Previous results have shown that our IEEE 802.15.1 standard was working perfectly with voice or even with data. That was a great help as one knew the IEEE 802.15.1 was correct and one could focus on the video compression. MATLAB has been used for this part of the project. MATLAB is a high-level technical language for algorithm development, data visualisation, data analysis and numerical computation, providing the user with immediate access to a wide range of analysis and design tools. These benefits make MATLAB the tool of choice for control systems, fuzzy logic control, neural fuzzy control, signal processing design, video and image processing, and communication system design [83].

In this section, one will focus on the video compression as the transmission and reception parts have been explained above. As explained in chapter 2, MPEG-4 is a video compression technique based on the following algorithm. Each block will be described in this section to enable the reader to understand the way the project has been built.

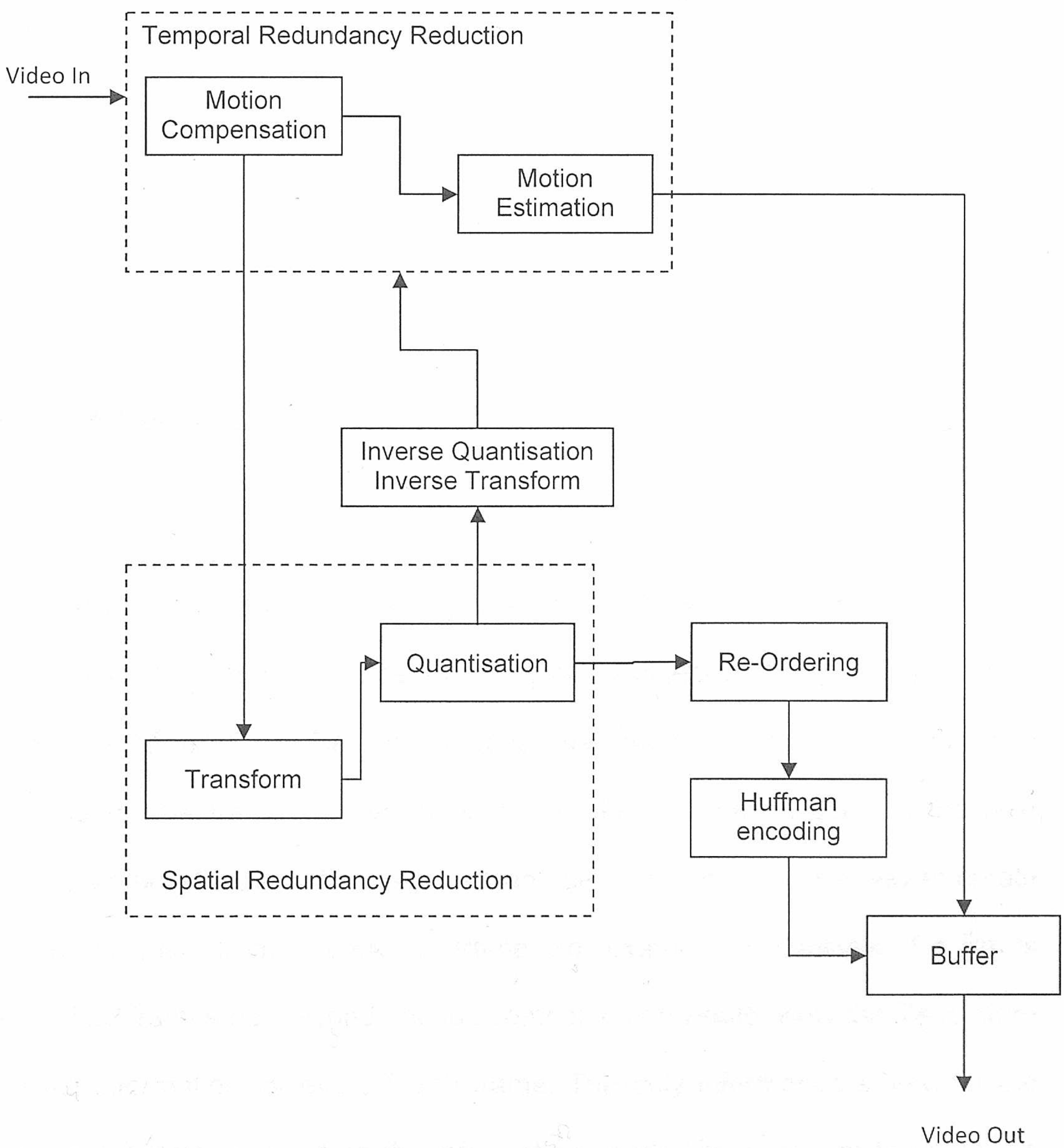

Figure 5.15: Block diagram

All the listed blocks are developed under:

- Motion Compensation
- Motion Estimation
- Transform and Inverse Transform
- Quantisation and Inverse Quantisation
- Huffman encoding
- Re-ordering.

- **Motion Compensation:**

For many frames of a film, the only difference between one frame and another is the result of either the camera moving or an object in the frame moving. In reference to a video file, this means that much of the information that represents one frame will be the same as the information used in the next frame. Motion compensation takes advantage of this to provide a way to create frames of a film from a reference frame. For example, in principle, if a film is shot at 25 frames per second, motion compensation would allow the file to store the full information for every fourth frame. The only information stored for the frames in between would be the information needed to transform the previous frame into the next frame. If a frame of information is 1 MB in size, then uncompressed, one second of this film would be 25 MB in size. Applying motion compensation, the file size for one second of the film can often be reduced to 8 MB, for typical video material. In MPEG, images are predicted from previous frames (P frames) or bidirectionally from previous and future frames (B frames). After predicting frames using motion compensation, the coder finds the error which is then compressed using the DCT and transmitted.

- **Motion Estimation:**

The temporal prediction technique used in MPEG video is based on motion estimation. The basic principle of motion estimation is that, in most cases, consecutive video frames will be similar except for changes generated by objects moving within the frames. In the case of zero motion between frames, it is easy for the encoder to predict efficiently the current frame as a duplicate of the prediction frame. When this is done, the only information necessary to transmit to the decoder becomes the syntactic overhead necessary to reconstruct the picture from the original reference frame. When there is motion in the images, the situation is not as simple. The way that motion estimation goes about solving this problem is that a comprehensive two-dimensional spatial search is performed for each luminance macroblock. Motion estimation is not applied directly to chrominance in MPEG video, as it is assumed that the colour motion can be adequately represented with the same motion information as the luminance. It should be noted at this point that MPEG does not define how this search should be performed.

- **Transform and Inverse Transform:**

This part of the program is based on the discrete cosine transform (DCT) (explained in chapter 2). And one is using a MATLAB function realise the DCT as explained below.

Y = dct(x) returns the unitary discrete cosine transform of x

$$y(k) = w(k)\sum_{n=1}^{N} x(n)\cos\frac{\pi(2n-1)(k-1)}{2N} \qquad k = 1, \ldots, N \qquad (5.1)$$

$$\text{where} \quad w(k) = \begin{cases} \dfrac{1}{\sqrt{N}} & k = 1, \\ \sqrt{\dfrac{2}{N}} & 2 \le k \le N \end{cases} \qquad (5.2)$$

N is the length of x, x and y are the same size. If x is a matrix, dct transforms its columns.

The DCT is closely related to the discrete Fourier transform. You can often reconstruct a sequence very accurately from only a few DCT coefficients, a useful property for applications requiring data reduction.

- **<u>Quantisation and Inverse Quantisation:</u>**

After transformation all of the information about the image and the whole amount of data is still held, no compression has yet been performed. However, the transformation causes the energy to be distributed in a more easily reducible way. For DCT coefficients, the maximum of energy is concentrated at the lowest frequency components and thus the majority of coefficients have little energy. By applying quantisation, not important values will be removed. The principle of quantisation is to divide the values by a non-zero positive integer (quantisation value) and round the quotient to the nearest integer. Uniform quantisers are usually defined by two parameters: the step q and the factor S.

- **Huffman encoding:**

Entropy encoding is used regardless of the media's specific characteristics. The data stream to be compressed is considered to be a simple digital sequence, and the semantic of the data is ignored. Here one is using a specific entropy encoding scheme called Huffman coding. It was developed by David A. Huffman, for computer science and information theory [84]. It is based on statistical methods. Given the character that must be encoded, together with the probability of their occurrences, the Huffman encoding algorithm determines the optimal code using the minimum number of bits. Hence the length (number of bits) of the coded characters will differ. In text, the shortest code is assigned to those characters that occur most frequently. To determine a Huffman code, it is useful to construct a binary tree. The nodes of this tree represent the characters that are to be encoded. Every node contains the occurrence probability of one of the characters belonging to this sub-tree; 0 and 1 are assigned to the edges of the tree. The two characters with the lowest probabilities are combined in the first binary tree. Their root node is labelled with these characters and the combined probability. The edges are labelled with 1 and 0 respectively. The nodes below this root node are not being considered further. Again, the two nodes with the lowest probabilities are combined into a binary sub-tree, the root node and the edges are labelled as before. Continue in this way until the node with the whole used alphabet and the probability 1 is reached. The encoded data for the characters are the paths through the tree to their nodes. The result is stored in a table. If the information of an image can be transformed into a bit stream, such a table can be used to compress the data without loss.

- **Re-ordering:**

The re-ordering box is an essential part of the codec. As the MPEG-4 has some hierarchy in the compression, obviously the I frames have to be coded and decoded before the other pictures, as the P and B frames are based on the I frames. The P frames have to be encoded and decoded second, as the B frames are based on the I and the P frames. This means that, during the process of coding and decoding, the order is not the order needed to send the GOP. For example, during the coding the process goes for the frame I1, then P4, I7, P10, I13, then B2, B3, B5, B6, B8, B9, B11and finally B12. As the correct order for a MPEG-4 video is IBBPBBIBBPBBI..., one needs to re-arrange the coding sequence in a correct MPEG-4 sequence.

Figure 5.16 shows the results for the No-AI model for a clip from the film "007-Tomorrow Never Dies" from Group Of Picture (GOP) 150 to GOP 200 at 724Kbps transmission rate.

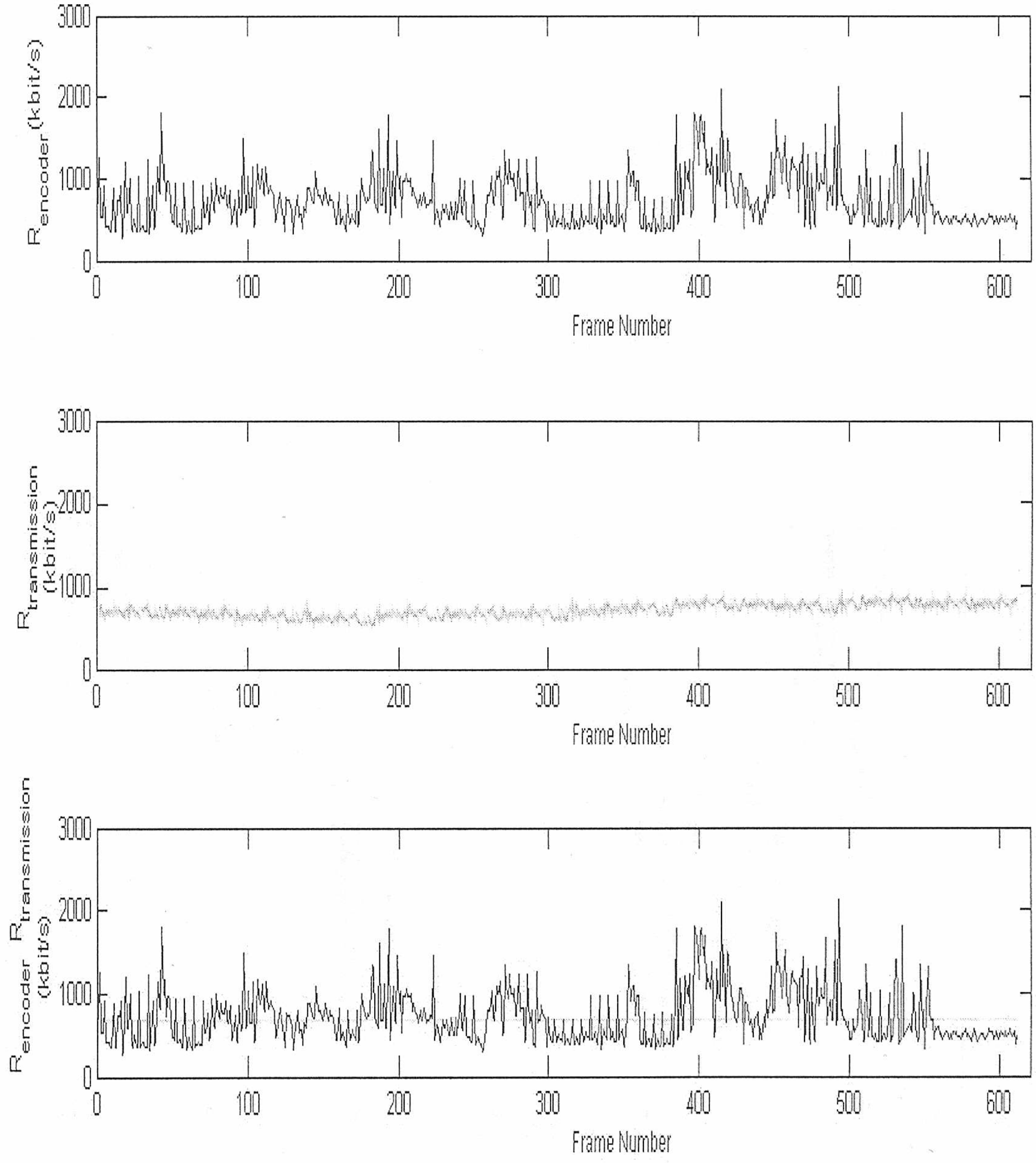

Figure 5.16: $R_{encoder}$, $R_{transmission}$, No AI system 007 GOP 150 to 200 724 Kbps

In Figure 5.16, three different graphics are visible. The one on the top of the figure is the signal's rate at the output of the MPEG-4 codec (Rencoder), just before the token bucket. The second signal is the transmission channel's rate of the IEEE 802.15.1 (Rtransmission) with noises and interferences. The third and last graphic is the aggregation of the signal sent (Rencoder) with the mean value decided at the beginning of the simulation, here 724 Kbits/s. In the third graphs, a large amount of data is over the mean values. That does not mean that all the data has been dropped, but it means the token bucket will be very well used and if any transmission problem occurs, the token bucket will not be able to hold more data and overflow will appear.

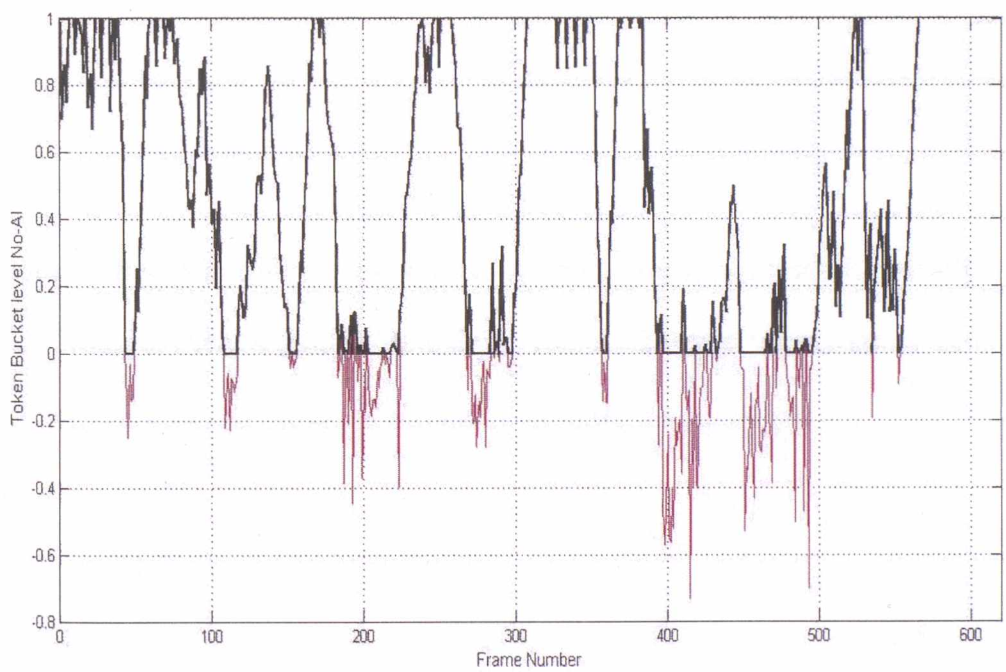

Figure 5.17: Token bucket availability for No-AI system 007 GOP 150 to 200 724 Kbps

Figure 5.17 explains the level of the token bucket during the transmission of the film "007 – Tomorrow Never Dies" from the group of picture 150 to the group of picture 200, with a bandwidth of 724Kbps. As one can see, the token bucket level is directly related to the rate at the reception point. For example, if one is looking between frame 400 and frame 500, the rate at the receiving part is mainly over the 724 Kbps maximum bandwidth all the time. Lots of data has to be stored in the token bucket in order to regulate the flow, unfortunately, the amount of data to be stored is too much for the size of the token bucket. Thus it appears that the values under 0 are really important and the result is the loss of data. In other words, when the value of the token bucket is under zero, data are lost and degradation of the quality of the picture appears. As the simulation is dealing with real-time transmission, data lost cannot be re-sent, that would create excessive delay.

Figure 5.18 shows the results on a No-AI model for a clip from the film "X-men" from Group of Picture (GOP) 50 to GOP 150 at 724Kbps transmission rate.

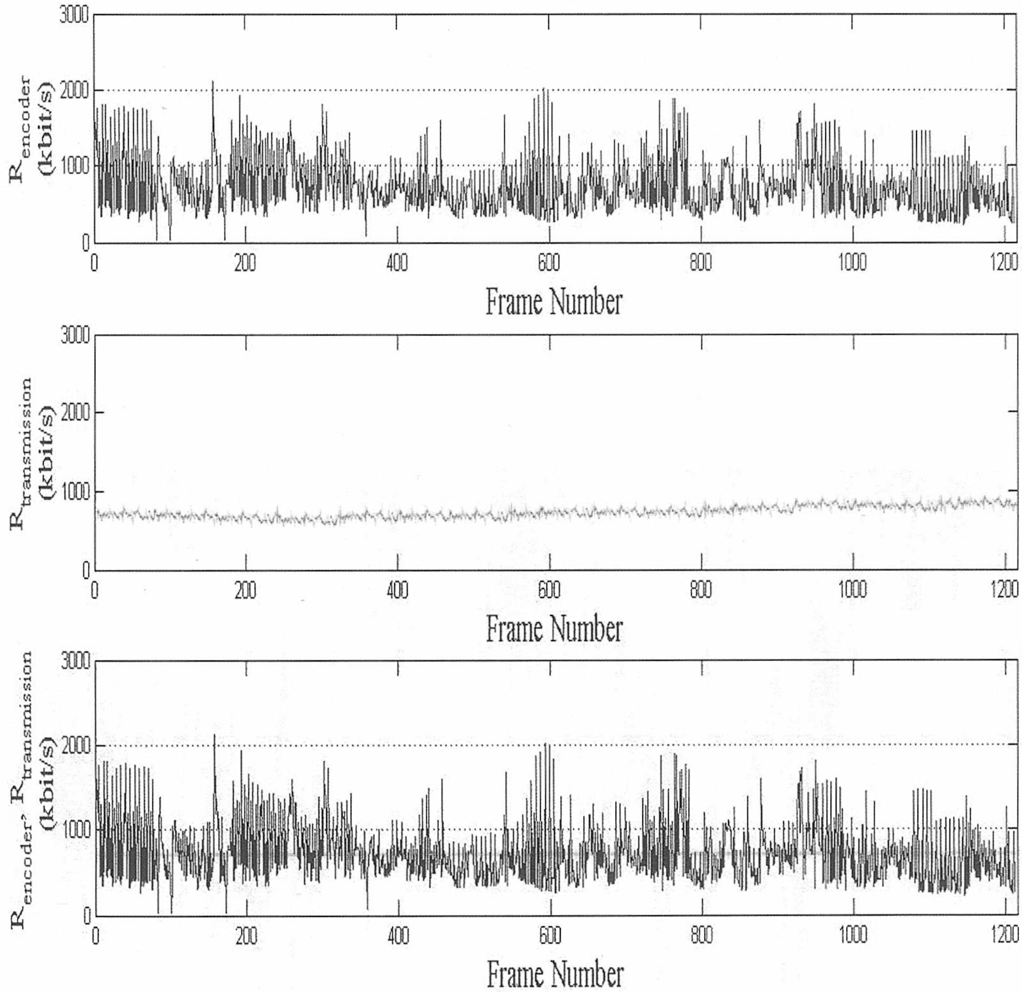

Figure 5.18: $R_{encoder}$, $R_{transmission}$, No-AI system X-men GOP 50 to150 724 Kbps

In Figure 5.18, one simulated the film "X-men" between GOP 50 and GOP 150, which means around 1200 frames. A Gaussian noise has been added to the mean value of 724Kbps as shown in the middle of the figure to simulate noises and interferences during the transmission, one can see a variable value of Rtransmission around the mean value. On the third and bottom graph, the aggregation of the two signals Rencoder and Rtransmission without the noise, one can easily see the mean value is mainly under the actual Rencoder value. That result engenders emptying the token bucket (full of data) and saturating the "added buffer". As explained in the fuzzy logic rules, the output becomes quickly at the point of overflowing, congestion occurs and data are lost.

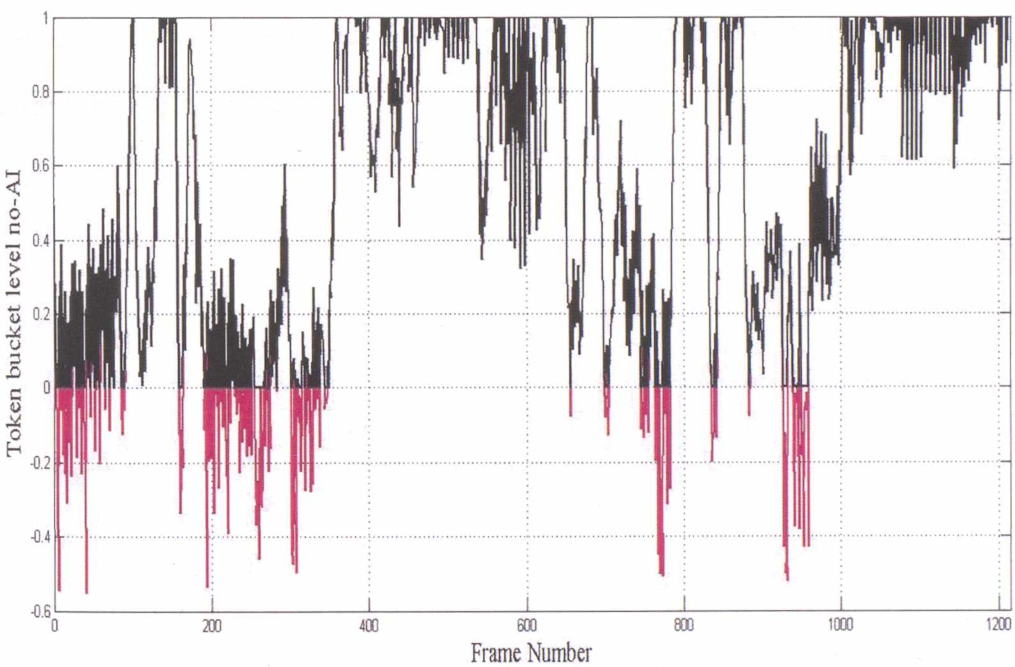

Figure 5.19: Token bucket availability for No-AI system X-men GOP 50 to 150 724 Kbps

By focusing on a particular point of Figures 5.18 and 5.19, one could explain and understand more easily what the token bucket is about. If one takes frames between 400 and 600 of the Rencoder and Rtransmission signals, as highlighted in the next figure (Figure 5.20) and compares it to the token bucket level of starvation, the similarities are obvious. During this gap (400 frames–600 frames), the data rate transmitted is mainly under the mean value of 724Kbps, so the data are going through without problem and the token bucket level is very close to one which means the maximum of space available for data. At the end of the gap studied here, the information is becoming heavier and the rate of data transmitted becomes higher and higher compared with the mean value. This has an effect of starving the token bucket by using the space available inside. One could see how data occupies the space in the token bucket just before the 600th frame.

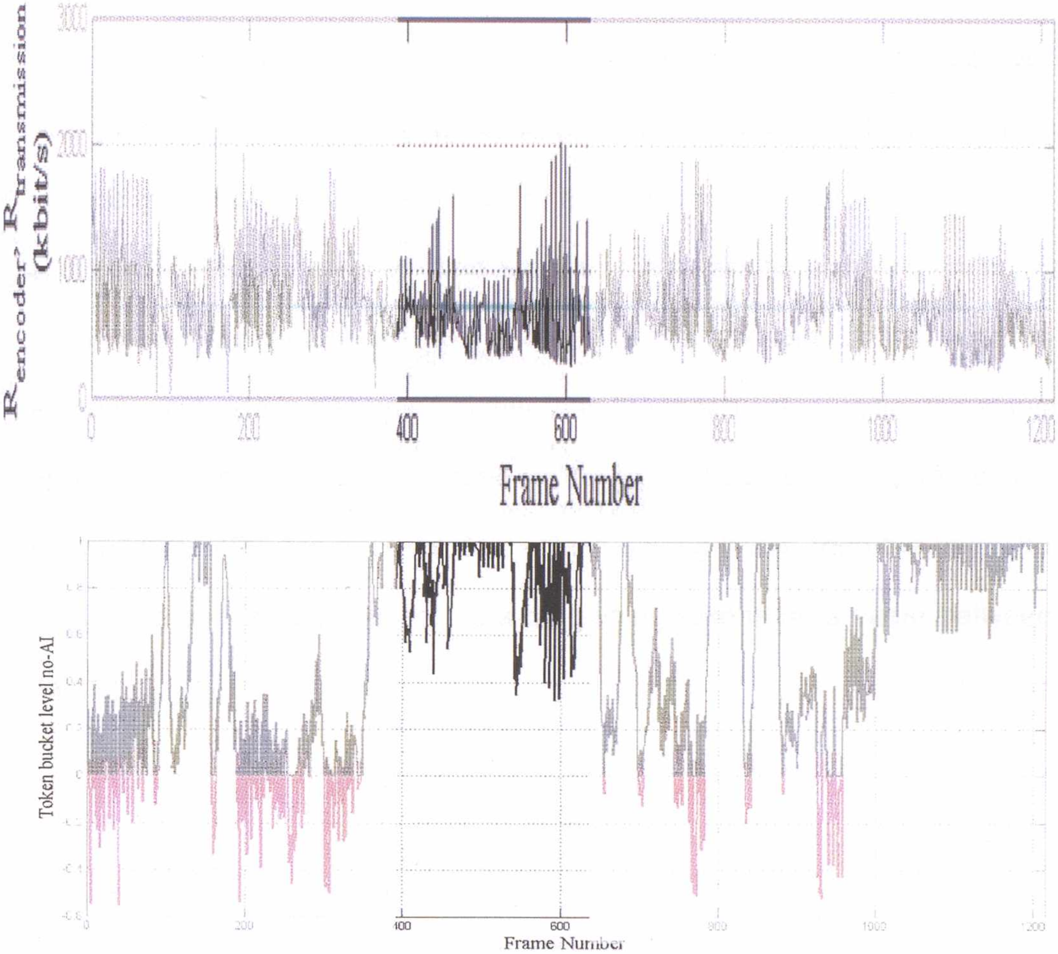

Figure 5.20: Highlighting the relation between the data rate level and the token bucket load, between frame 400 and 600.

On the same figure but at a different frame, a problem occurs as the data rate is becoming much higher than the mean value. Let us show the window one is interested in on this example. Here one is looking at frames from 200 to 400, and the average data rate is higher than the mean value, as shown on the first part of Figure 5.21. On the second part of the graph, one could see the starvation, even more the overflow of the token bucket (all the values under zero are lost or delayed). The correspondence between the data rate over the virtual limit of the mean value of 724Kbps and the starvation of the token bucket is once more obvious. At the middle of the gap one are studying, one could see a diminution of the level of data sent, in the exact time the token bucket is gaining back some tokens and the overflow has stopped. But as soon as the average data rate is becoming higher than the mean value again, the starvation appears one more time as exposed on the second part of the figure below. Just before frame 400, the average data rate is getting lower, then the token bucket gets more space available for data by gaining tokens as a value between 0.7 and 1.

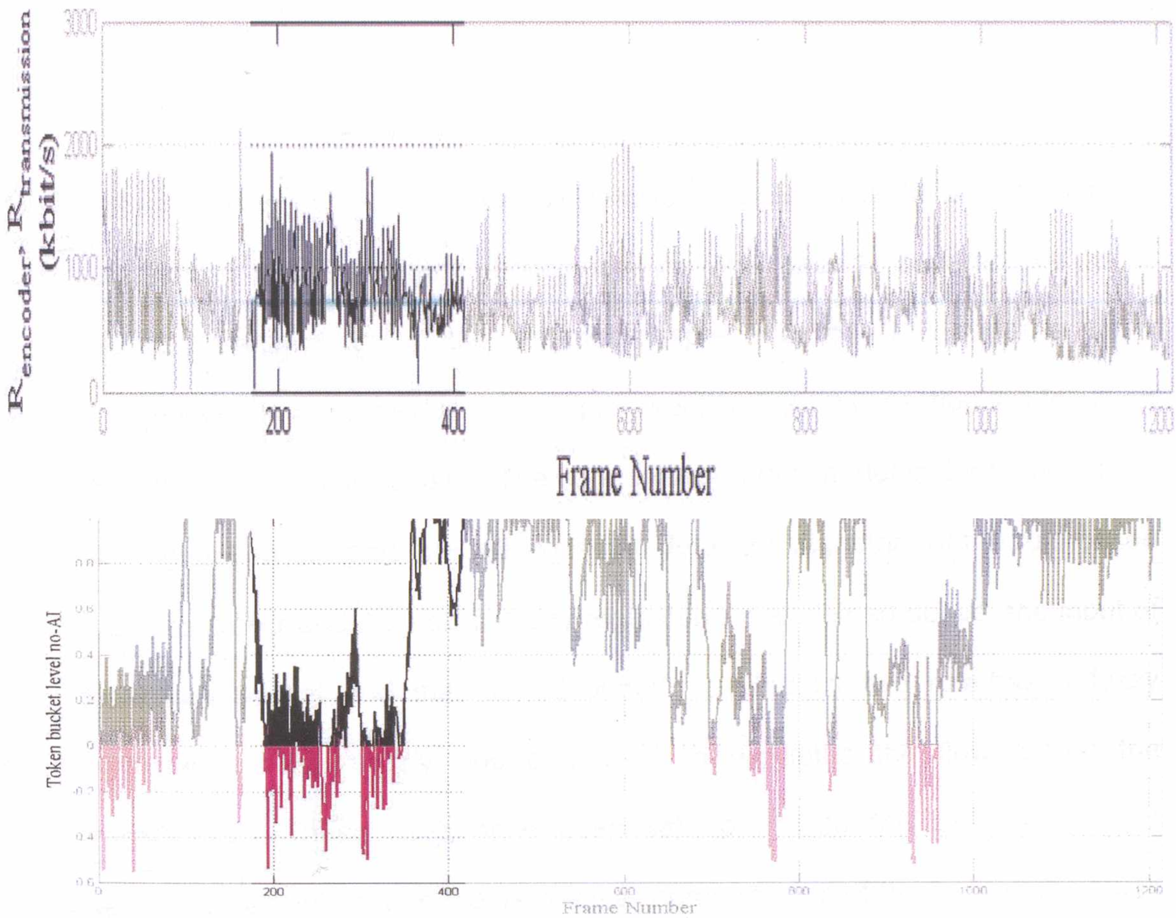

Figure 5.21: Highlighting the relation between the data rate level and the token bucket load, between frame 200 and 400.

5.3 Video transmission over IEEE 802.15.1 standard with intelligent technique

In the following part, one will explain results simulated with the second design. This model uses intelligent techniques. As one already covered the IEEE 802.15.1 transmission in the first sub-chapter (section 5.1) and the video compression in the previous sub-chapter (section 5.2); this section will focus on the intelligent technique used. The codec used in this model is the same as the VBR model. In this part one will explain in detail the algorithm of the rule-based fuzzy controller and the neural fuzzy regulator. As explained above, the input of the token bucket and the "added buffer" is ruled by two rule-based fuzzy controllers to have only one output which regulates the flow before the transmission. These rules have been set up on the MATLAB fuzzy logic toolbox. One could see in Figure 4.9 the representation of the level of the token bucket membership and in Figure 4.8 the level of the "added buffer" membership.

Figure 5.22 shows the results for a clip from the film "007 – Tomorrow Never Dies" from Group Of Picture (GOP) 150 to GOP 200 at 724Kbps transmission rate (the value decided previously as the maximum bandwidth available with the IEEE 802.15.1 standard).

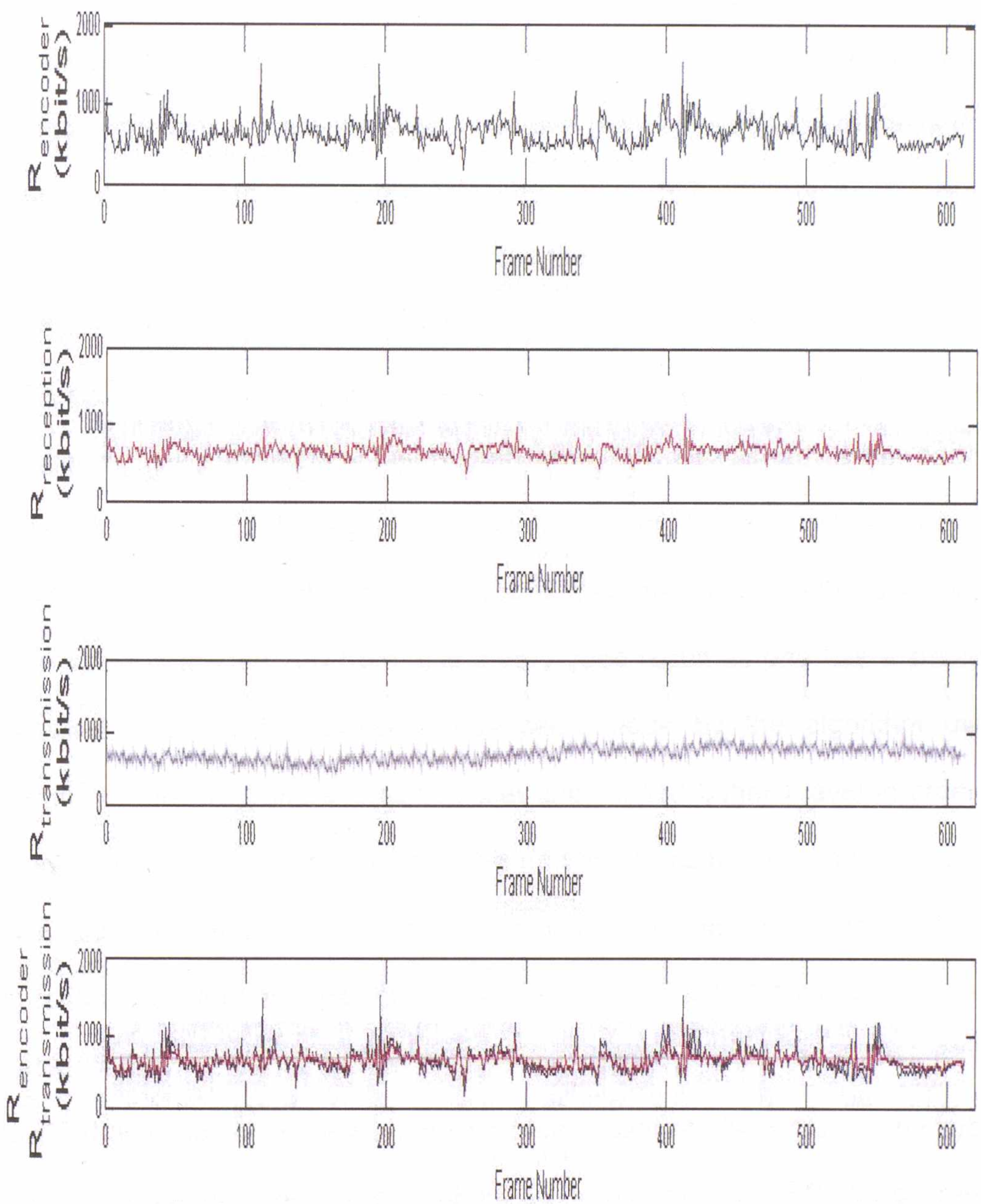

Figure 5.22: $R_{encoder}$, $R_{reception}$, $R_{transmission}$, AI system 007 GOP 150 to 200 724 Kbps

Figure 5.22 contains a representation of the three signals; Rencoder, Rreception and Rtransmission. Rencoder is, as previously explained, the signal at the end of the decoder (just after the MPEG-4 encoding). On the top part of the figure, the signal has very large amplitude (value in Kb/s) the range is from around 500 Kb/s minimum to a maximum of more than 1500Kbps. This signal (Rencoder) receives no modification from the system, one is using it as the reference to compare before and after this system to evaluate the performance. In the second part of the figure is the signal Rreception, this signal represents the usage of the bandwidth at the receiving end. One can already easily see the smaller amplitude of the signal. Effectively, the amplitude is still around 500 Kb/s for the minimum value but the maximum is rarely over 1000 Kb/s. As the mean value is around 720 Kb/s, it is a very good result as one has a smallest bandwidth usage. This evolution has been made by the algorithm using analysis of the two buffers (token bucket and "added buffer") level in order to smooth the information in case of problem during transmission. In the following table, one could find the proof of the amplitude reduction between the hybrid neural-fuzzy system and the VBR system. Values in Table 5.1 show a reduction of at least thirty percent between the learning system and the usual design. The standard deviation here varies around a mean value of 724 Kbps, for the hybrid system the standard deviation is about 93 Kbps and the VBR design has a standard deviation of 322 Kbps. This information is really important in term of bandwidth efficiency as one is proving here that the design reduce the standard deviation which is going to help to decrease the data loss and then will improve the quality of picture at the receiving end. The prevention of data loss is one of the most important aspects to deal with in order to obtain a better QoS.

	AI SYSTEM	NO-AI SYSTEM
Standard deviation for a mean value around 724 Kbps	93.6666Kbps	322.2758Kbps

Table 5.1: Standard deviation comparison between AI-system and No-AI system 007 GOP 150 to 200 around 724 Kbps.

The third part of the Figure 5.22 is the transmission rate of the IEEE 802.15.1 (Rtransmission) with noises and interference. Here and for all the tests one fixed the mean value at 724Kbps. The last graph of the figure is the superposition of the Rencoder, Rreception and the mean value without noise (used to see the exact position of the maximum bandwidth available). In this fourth part of the figure, one can compare the Rencoder and Rreception signals to highlight the reduction of the bandwidth usage. A more visual figure of the result will be developed in the last part of this section to show more easily the relation between the bandwidth usage and the buffers usages. The buffers usages are shown in the following Figure 5.23.

Figure 5.23 shows the amount of data stocked in the different buffers. The top part of the figure shows the quantity of the data put on the "added buffer" during the simulation time. From the source is group of picture 150 to group of picture

200 for the “007” video clip. On the second part of the figure, one has the quantity of token still in the bucket. The correspondence of the usage in Figure 5.22 and Figure 5.23 will be highlighted in Figure 5.24 and Figure 5.25.

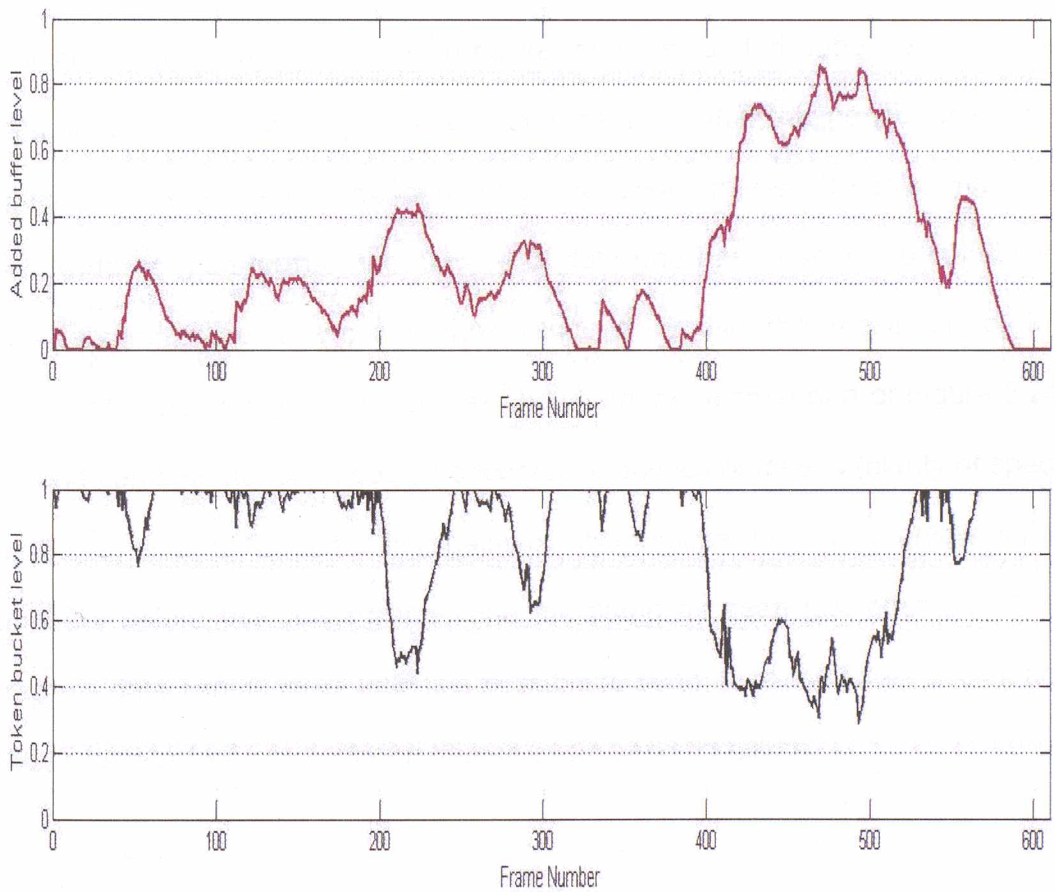

Figure 5.23: Buffers availability for AI system 007 GOP 150 to 200 724 Kbps

To understand how these figures work together, one has taken the last graph from Figure 5.24 and all of Figure 5.25 and highlighted some parts. For the first analysis, one has highlighted the range between 200 and 300 frames. In this range, one could see on the top of Figure 5.24 the signal of the Rtransmission, the signal of Rencoder and the mean value without noise. This last would be the value one is looking for to see if too much information is sent. Aligned with this graph, one added the buffers usage. One can easily see that values after the 200th frame are over the limit of 724 Kbps fixed at the beginning of the project. In concordance, the token bucket and the "added buffer" are stocking data as too much information is needed to be sent. After this period, one has a space with less data to be sent near the 250th frame. On this area, the bandwidth is greater than the amount of data one has to send, so controllers are sending more information than required to use the maximum of the bandwidth and empty the buffers, as shown on the second part of Figure 5.24. As one can see the token bucket becomes nearly full of tokens (plenty of space available) and the "added buffer" gains plenty of space. Another overflow period appears before the 300th frame, and the same as seen just after the 200th frame – data sent is more than it is possible to send, then the token bucket and the "added buffer" are buffering data to avoid sending failure.

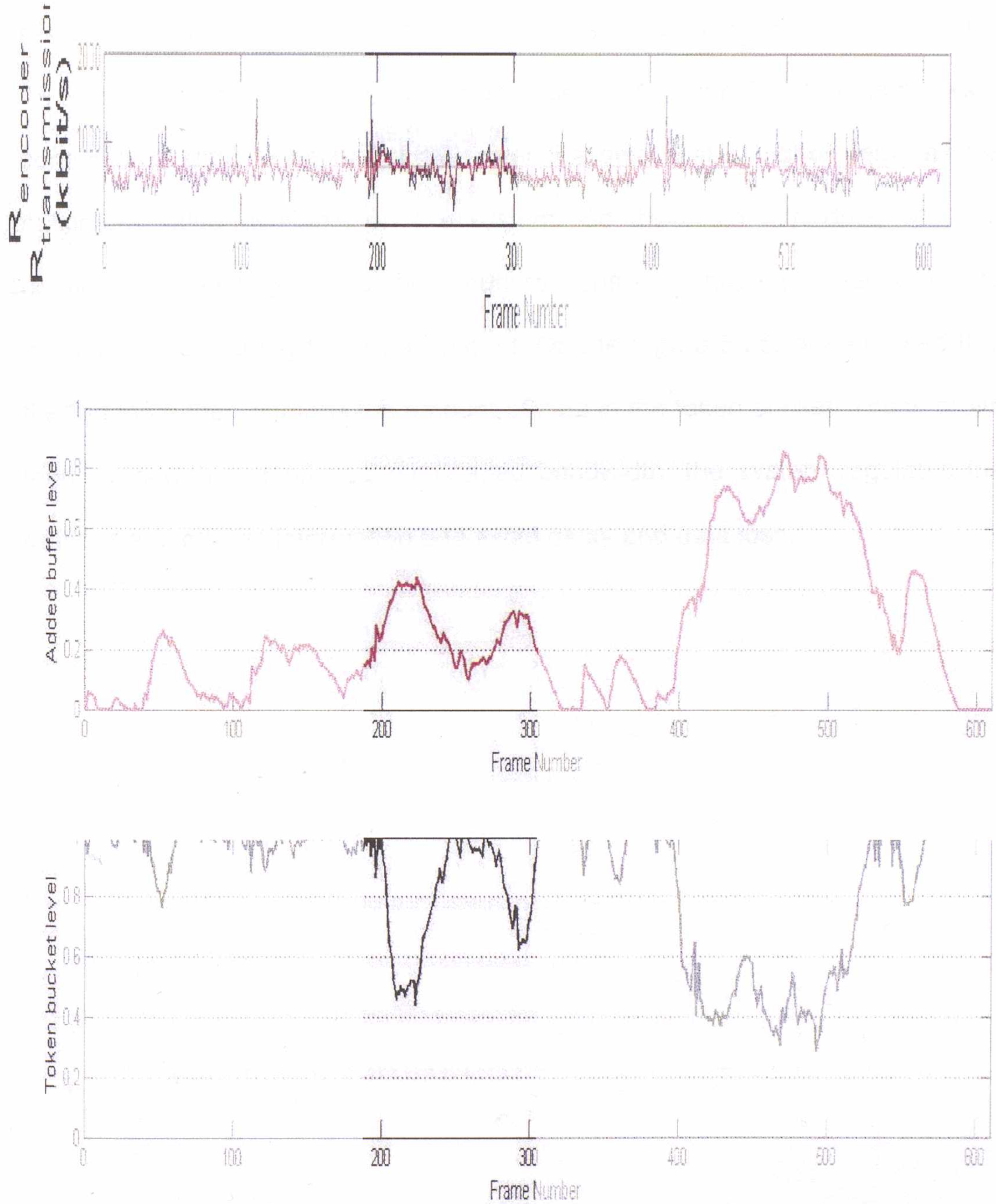

Figure 5.24: Highlighting the relation between the data rate level and the buffers load, between frame 200 and 300.

In Figure 5.25 one highlights the frames between 400 and 500 as one has another interesting evolution of the buffers (token bucket and "added buffer"). As already explained above, the range between 400 and 500 frames shows a continuity of data to be transmitted over the device at a higher value as the maximum value available on the system. So the main difference with the previous highlighting is that both buffers constantly have to deal with data arriving and departing to be transmitted. On the Figure 5.25, one can see that the "added buffer" lightens the amount of data in the token bucket. These prove than even with a continued overloaded bandwidth, the system regulates the traffic to smooth the information and avoid delay and data loss.

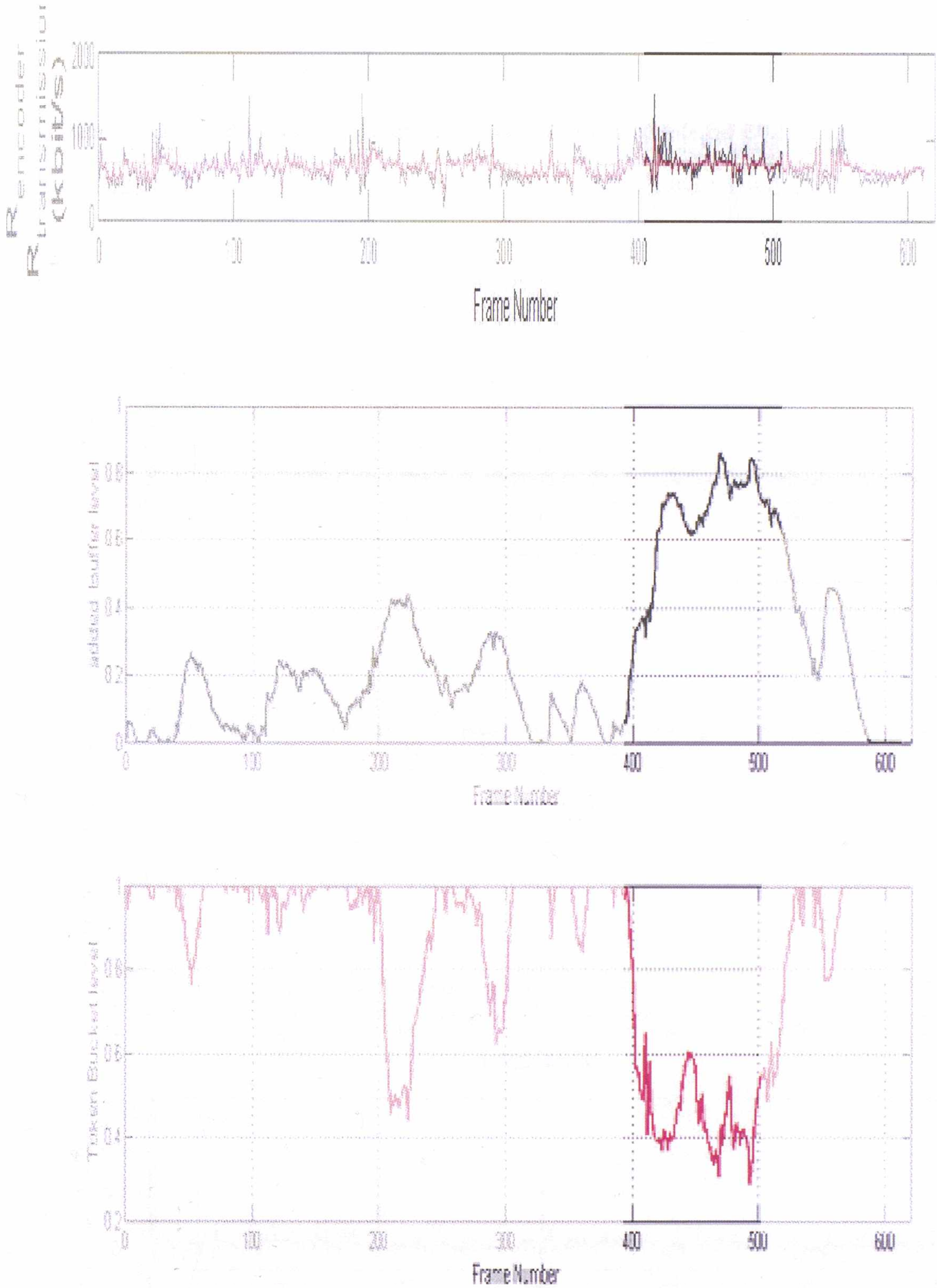

Figure 5.25: Highlighting the relation between the data rate level and the buffers load, between frame 400 and 500.

Figure 5.26 shows results for a clip from the film "X-men" from Group Of Picture (GOP) 50 to GOP 150 at 724Kbps transmission rate (the value decided previously as the maximum bandwidth available with the IEEE 802.15.1 standard).

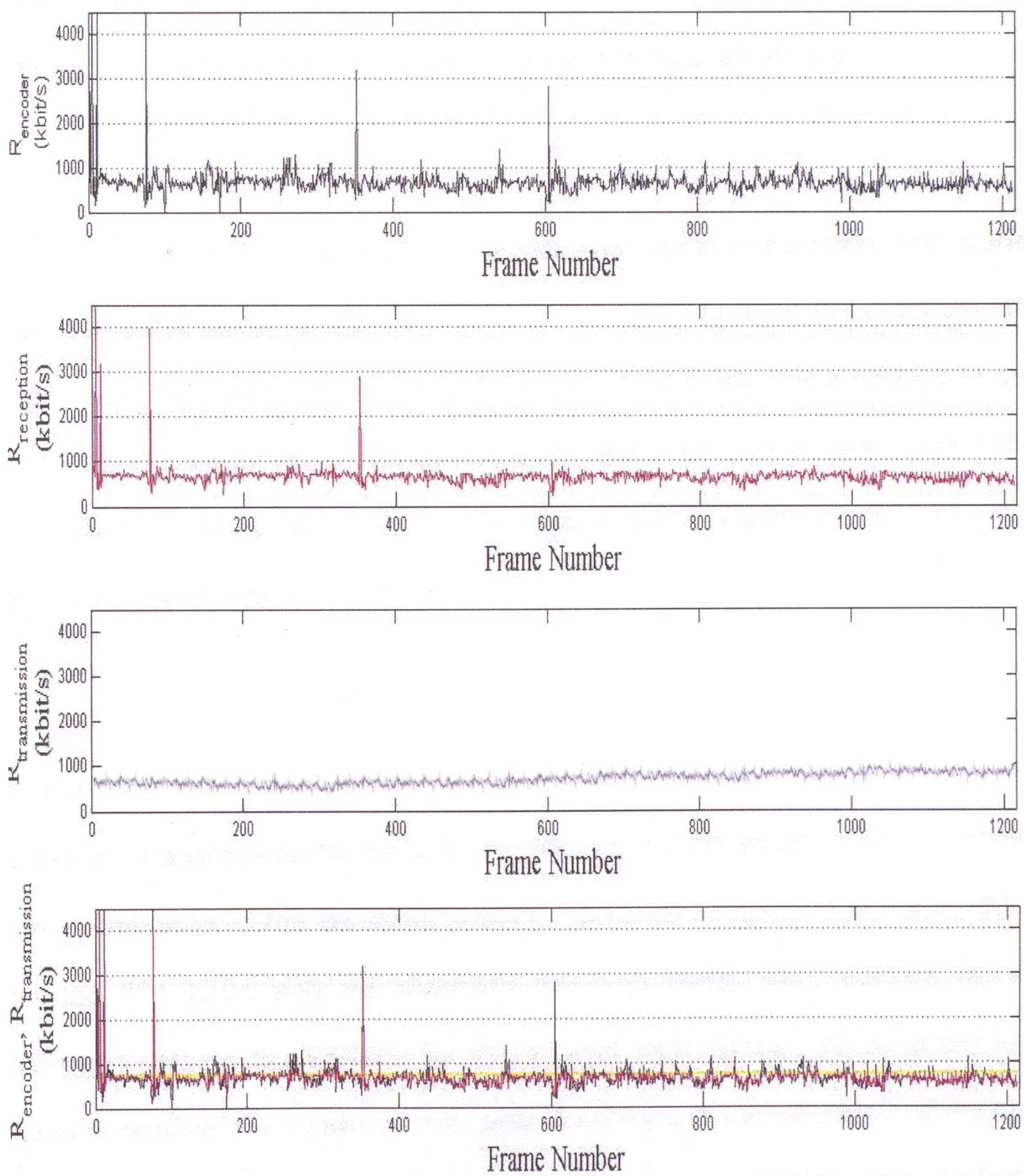

Figure 5.26: $R_{encoder}$, $R_{reception}$, $R_{transmission}$, AI system X-men GOP 50 to 150 724 Kbps

Figure 5.26 is a representation of the three signals; Rencoder, Rreception and Rtransmission. Rencoder is, as previously explained, the signal at the end of the decoder (just after the MPEG-4 encoding). In the top part of the figure, the signal has very large amplitude (value in Kb/s); the range is around 500 Kb/s minimum and one has some maximum peaks at more than 4000Kb/s, but mostly the signal is just under 1000Kb/s and often overtakes this value. This signal (Rencoder) receives no modification from the system, one is using it as a reference to compare before and after the system, to evaluate the performance.

In the second part of the figure, one has the signal Rreception, this signal represents the usage of the bandwidth at the receiving end. One can already easily see the smaller amplitude of the signal. Effectively, the amplitude is still around 500 Kb/s for the minimum value, but the maximum is rarely over 1000 Kb/s. As the mean value is around 720 Kb/s, it is a very good result as one has a smallest bandwidth overflow.

In the table 5.2, a comparison of the standard deviation during the "X-men" video clip transmission for the learning system and the common system shows the optimization of the bandwidth usage. The mean value of the bandwidth is around 724 Kbps and the standard deviation varies from 83Kbps for the intelligent design to 296Kbps for the original VBR design. An amazing forty percent reduction is shown on this table confirming the amelioration of the QoS in term data loss prevention. The hybrid system here optimises the bandwidth usage to prevent data loss then increase the efficiency. This results in a better quality of picture at the receiving end.

	AI SYSTEM	NO-AI SYSTEM
Standard deviation for a mean value around 724 Kbps	89.34Kbps	296.57Kbps

Table 5.2: Standard deviation comparison between AI-system and No-AI system X-men GOP 50 to 150 724 Kbps.

This evolution has been made by the algorithm that was used to analyse the two buffers (token bucket and "added buffer") level in order to smooth the information in case of problems appearing during transmission. The third part of the Figure 5.26 is the transmission rate of the IEEE 802.15.1 (Rtransmission) with noises and interference. The last graph of the figure is the aggregation of the Rencoder, Rreception and the mean value without noise (used to see the exact position of the maximum bandwidth available). In this fourth part of the figure, one can compare the Rencoder and Rreception signals to highlight the reduction of the bandwidth usage.

In this set of graphs, one can see a very important problem. Some peaks appear; two are located at the very beginning of the simulation, another one is around the 100th frame, a third one is around 350th frame and the last one is on the 600th frame. Those enormous peaks create a very high demand on the IEEE 802.15.1 device to stock quickly a very large amount of data. Both of the buffers (token bucket and "added buffer") have been used for these peaks, as is described in Figure 5.27. But a very characteristic point here is the last peak shown on the Rencoder signal, because the system completely drops the peak on the Rtransmission signal.

One can follow the evolution of the availability of the buffers in Figure 5.27. In this figure, one has an example of the possibility of overflow even with the system. At the beginning of the transmission, a few peaks of data arrived and the algorithm was not able to treat completely all the information and that created an overflow on the token bucket as well as the "added buffer", during this period data has been lost. One also needs to discuss two other significant points; one is located near the 100th frame, and the second around the 350th frame. The first point to be discussed shows an overflow of the token bucket, but not of the "added buffer", this is because of a very short peak of data sent. The system does not even have the time to see the overflow coming before it has happened, but one can see after the frame, a huge response of the "added buffer" which stocks a lot of data in a very short time. One has the same event and the same result at frame 350. One can see a small overflow of the token bucket and a huge response of the "added buffer", having its available space from a value near 0.8 to a value under 0.4 in a very short time (less than 20 frames). A more visual figure of the result is developed below to show more easily the relation between the bandwidth usage and the buffers usages.

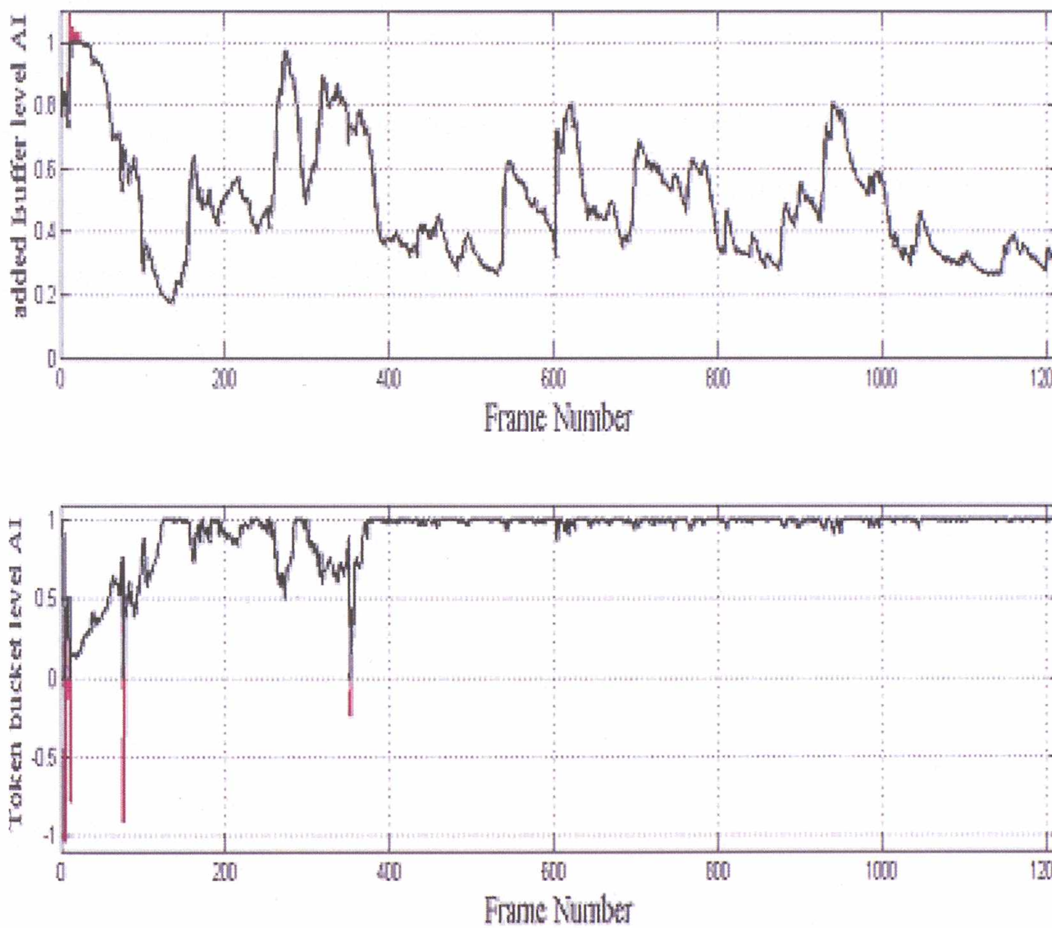

Figure 5.27: Buffers availability for AI system X-men GOP 50 to 150 724 Kbps

In the next figures, one highlights the frames that mentioned before. Four specific points have to be analysed, first at the very beginning of the simulation where an overflow of both of the two buffers appears, the second one around the 80th and the 350th frames, where only the token bucket overflows, and finally near the 600th frame, where the peak is totally eliminated. In Figure 5.28, one can highlight and zoom in on the very beginning of the simulation up to the 50th frame. At this part of the simulation one can see that even with the system overflow can happen. Looking more closely at the rate value of the Rencoder and Rtransmission, their signals have peaks nearly seven times higher than the maximum bandwidth available (724Kbps) for Rencoder. The system is not able to treat that amount of information so data is lost during this period which generates degradation of the quality of picture.

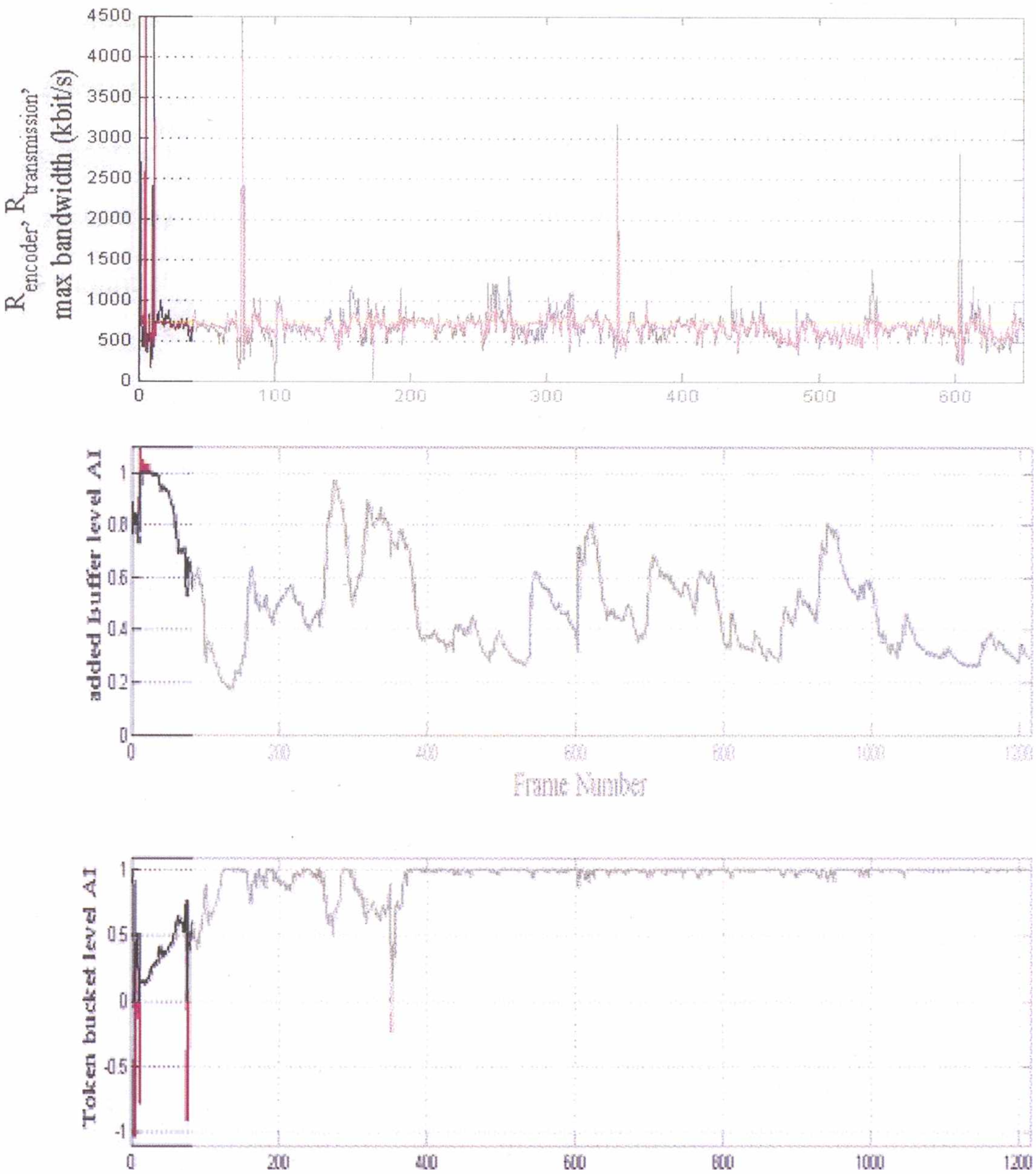

Figure 5.28: Highlighting the relation between the data rate level and the buffers load, at the very beginning.

The problem of overflow is also present at two other points of the token bucket, around the 70th and the 350th frames (Figure 5.29). These problems appear as the system is learning about the transmission rate, which means it needs time to analyse and treat the information in order to give an accurate answer. Unfortunately here the system has two very tight and short peaks, bringing loads of data to the token bucket without the possibility of the system sending a proper answer. But one can see, too late obviously, the response sent by the system by using the "added buffer" right after the peak and stocking a large amount of data (near the 100th frame and near the 370th frame).

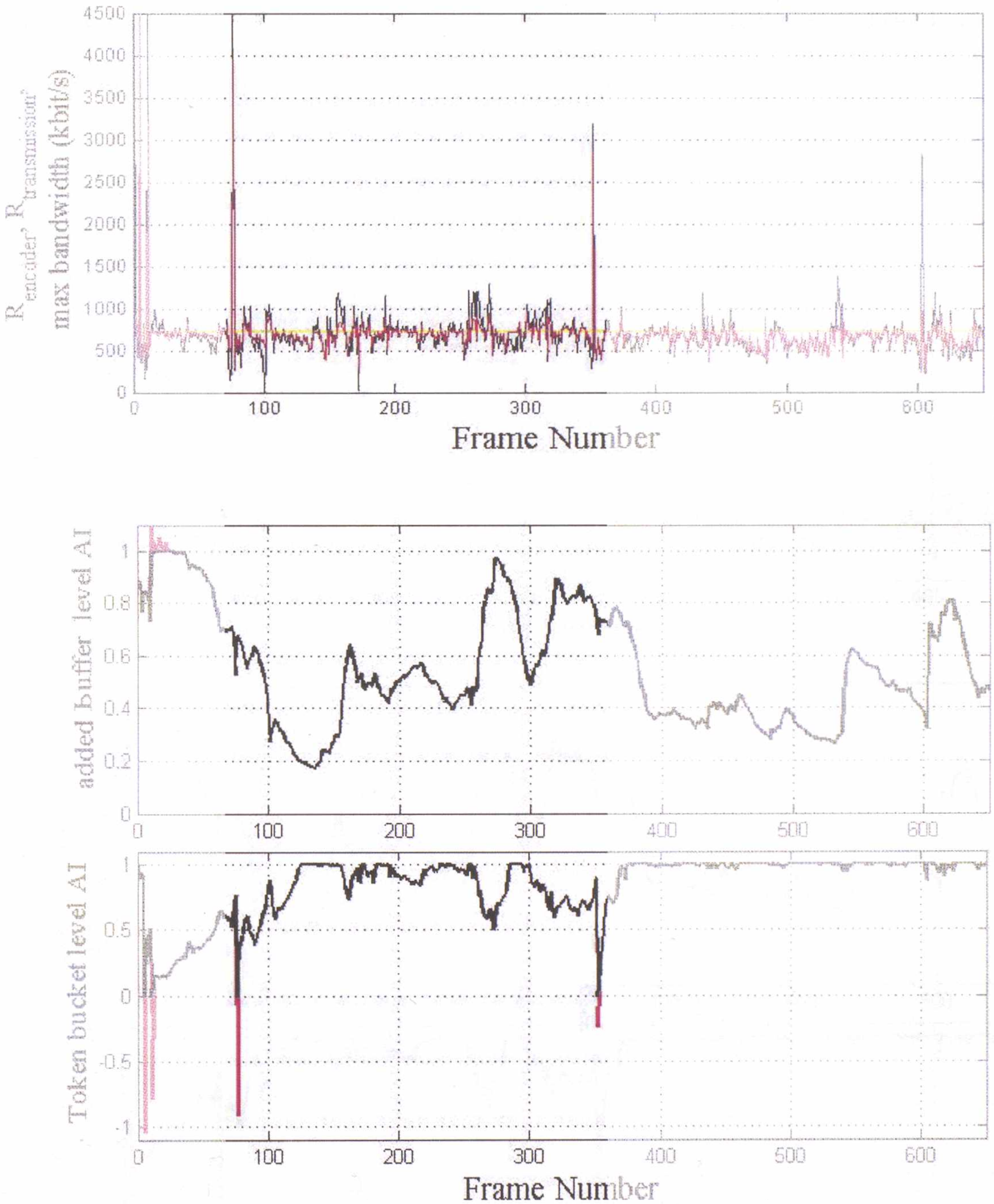

Figure 5.29: Highlighting the relation between the data rate level and the buffers load, between frames 80 to 350.

However, near the 600th frame (Figure 5.30), the system is able to answer the peak by using the "added buffer" and anticipate another overflow of the token bucket. It is really obvious that the space available in the “added buffer” decreases drastically while the token bucket remains full of tokens and ready to keep data if a problem occurs during the communication.

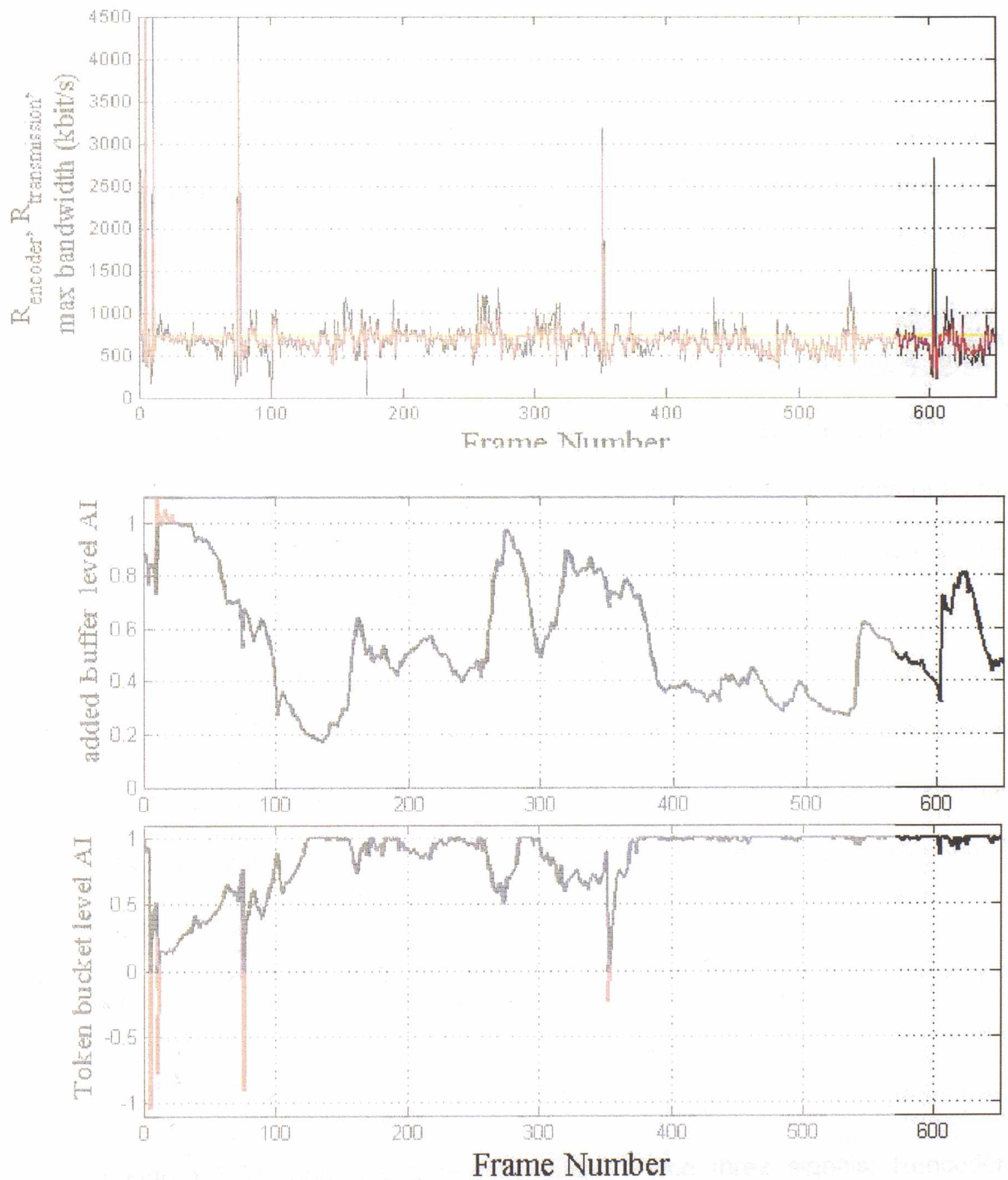

Figure 5.30: Highlighting the relation between the data rate level and the buffers load, around frame 600.

Figure 5.31 (below) shows results for a clip from the film "007 – Tomorrow Never Dies" from Group Of Picture (GOP) 100 to GOP 103 at 724Kbps transmission rate, with no noise added.

The system, in theory, analyses the bandwidth available in order to regulate the traffic and optimise the data sent over the device. As the system is based on adaptive technology, a postponed answer is made to regulate the traffic. This problem cannot be avoided as the neural network has to be trained before sending an accurate response. The response of the design can arrive a bit late if the peak of information is really high and tight, as shown on Figures 5.28 and 5.29, and that could create some degradation of the quality of picture. Despite this problem, the solution is showing very good improvement in terms of token bucket and bandwidth usage. All the results show a real improvement of the token bucket usage as well as the bandwidth utilisation, due to the action of the neural network and the fuzzy logic controllers. Even when noise and interference appears during the transmission, my system stocks more data in the "added buffer" and the token bucket to avoid data loss during the communication. By decreasing the quantity of data dropped due to overflow, I am increasing the quality of the picture.

In Figure 5.31, one has a representation of the three signals; Rencoder, Rreception and Rtransmission. Rencoder is, as the previously explained, the signal at the end of the decoder (just after the MPEG-4 encoding). In the top part of the figure, the signal has very large amplitude (value in Kb/s) the range is around 500 Kb/s minimum to a maximum of more than 1500Kb/s. This signal (Rencoder) receives no modification from the system, one is using it as the reference to compare before and after the system to evaluate the performance. In the second part of the figure, one has the signal Rreception, this signal represents the usage of the bandwidth at the receiving end. One can already easily see the smaller amplitude of the signal. Effectively, the amplitude is still around 500 Kb/s for the minimum value but the maximum is rarely over 1000 Kb/s. As the mean value is around 720 Kb/s, it is a very good result as one has the smallest bandwidth usage. This evolution has been made by the algorithm that was used to analyse the two buffers (token bucket and "added buffer") level in order to smooth the information in case of problem during transmission. The third part of the figure is the transmission rate of the IEEE 802.15.1 (Rtransmission) with noises and interference. Here, one did, the mean value was 724Kb/s. The last graph of the figure is the superposition of the Rencoder, Rreception and the mean value without noise (used to see the exact position of the maximum bandwidth available). In this fourth part of the figure, one can compare the Rencoder and Rreception signals to highlight the reduction of the bandwidth usage.

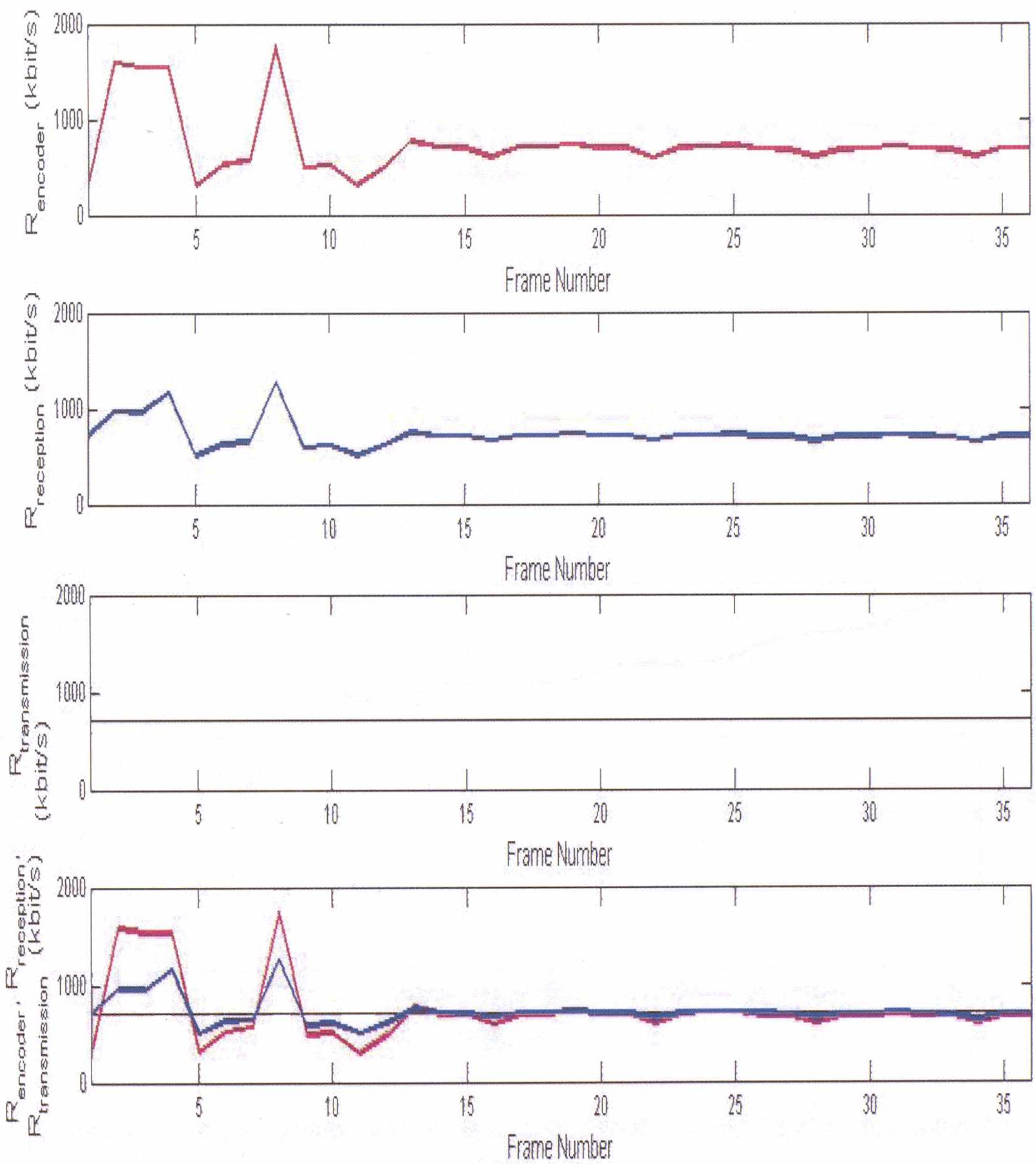

Figure 5.31: $R_{encoder}$, $R_{reception}$, $R_{transmission}$, AI system 007 GOP 100 to 103 724 Kbps, without noise.

Figure 5.32 gives a more visual figure of the result to show more easily the relation between the bandwidth usage and the buffers usages. The buffers usages are shown in Figure 5.32.

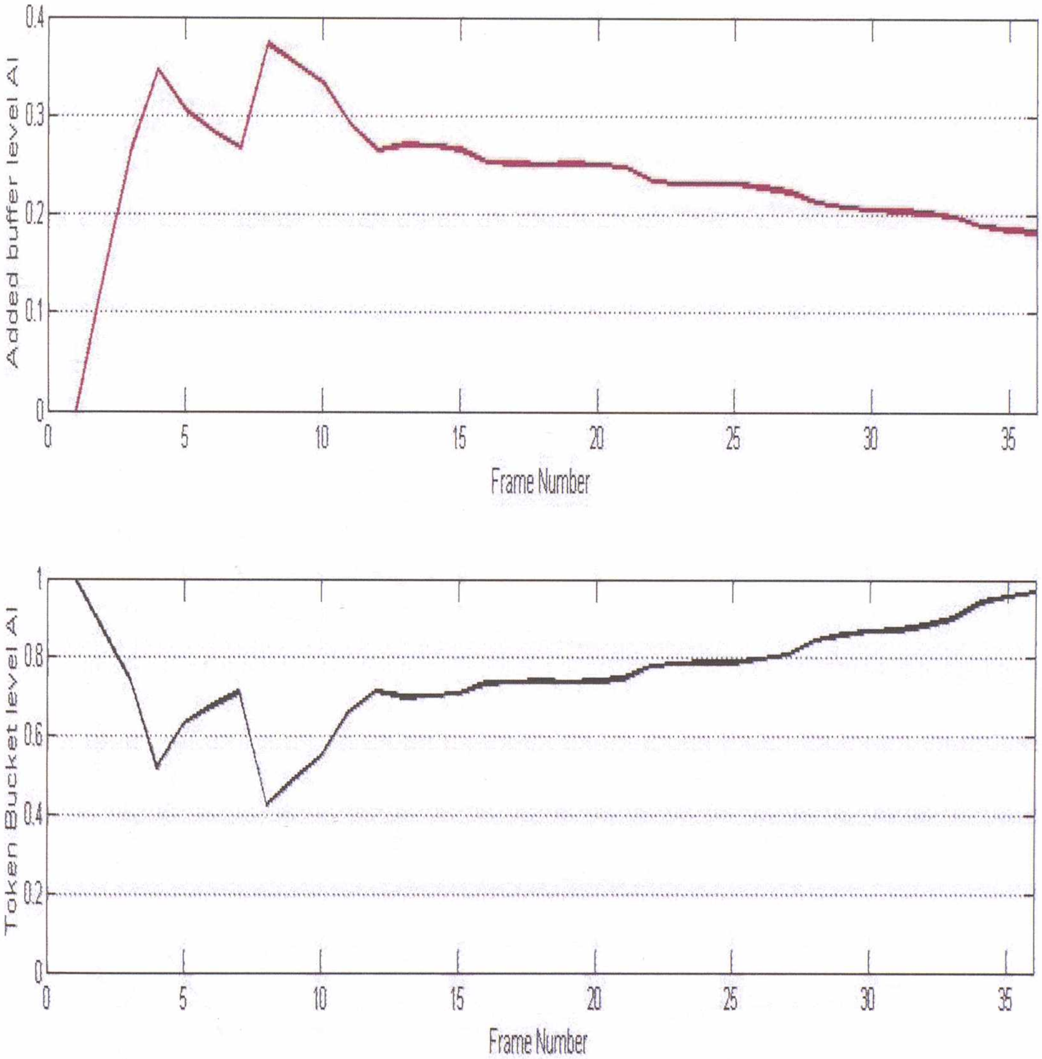

Figure 5.32: Buffers availability for AI system 007 GOP 100 to 103 724 Kbps without noise.

In the next figure, one highlights the frames between 0 and 10 as one has another interesting evolution of the buffers (token bucket and “added buffer”).

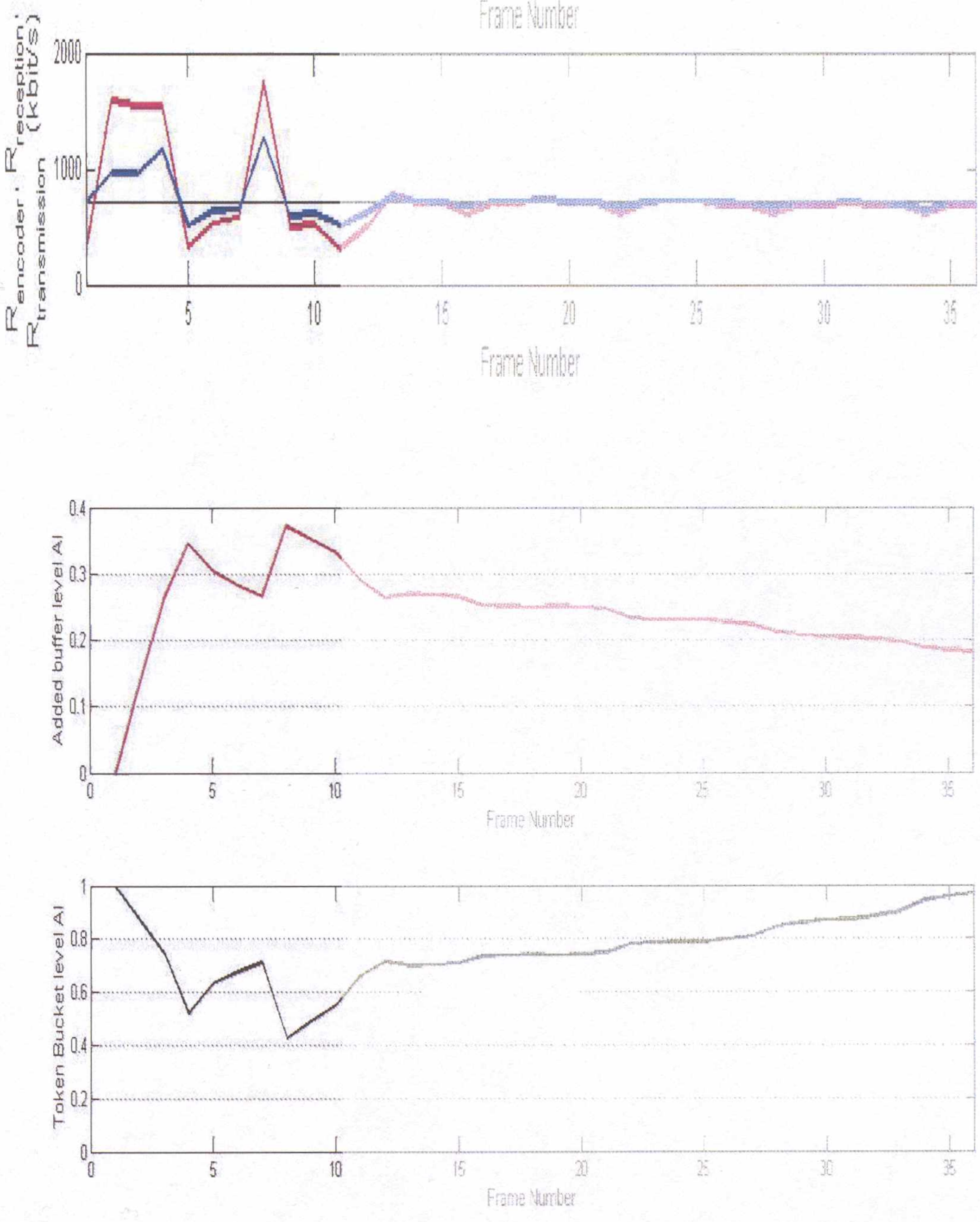

Figure 5.33: Highlighting the relation between the data rate level and the buffers load without noise, up to frame 10.

In Figure 5.33 one can see the correspondence between the rates into the system and the availability of buffers. This figure is based only on three groups of picture, which means only 36 frames, but it is really to see and understand how the system works and the stretch of the data on all signals.

CHAPTER 6:

Discussion, conclusion, contributions and recommendation

6.1 Discussion

The expansion of new methods of communication generated massive interest in wireless connectivity. Users require more and more options on their portable devices, for example, it is now natural to take a picture on your mobile phone or even to make a video film on the same mobile phone. Nowadays, one can even watch streaming TV programmes on our mobile phones. Based on this, manufacturers have to allow their mobile devices to transmit and receive these multimedia files with a sufficient quality of picture displayed on the device's screen. To do so, it is necessary to enlarge the bandwidth or change the compression and use more compressed data to fit on a small bandwidth. As for mobile device size, weight and autonomy is crucial, the bandwidth cannot be too large as it is going to use too much power and then it will not be power efficient. For the compression technique also, it is not so easy to compress any more than the compression used at the moment. Actually one could find some more accurate compression than the MPEG-4, but those techniques are still in development and need some improvement to be used properly. So where can one improve this "equation"?

During the entire project the aim was to increase the quality of picture and to reduce excessive delay on video transmission using the short-range IEEE 802.15.1 wireless standard. Previous researches had shown that improvements in picture quality are possible by adding intelligent technique. On this basis, one explored the idea of applying intelligent technique on wireless short-range network to transmit video. Fuzzy logic has been applied on multimedia support in order to improve the quality of the reception. Fuzzy techniques have been used in wireless connection such as Wi-Fi or Bluetooth. Neural networks have also been used in wired and wireless communication in order to reduce loss and data transmission problems. However, implanting artificial intelligence in IEEE 802.15.1 standard devices in order to increase the quality of picture by reducing the data loss and excessive delay has not been attempted before. For this project, one decided to use MPEG-4 video compression. As explained previously, MPEG-4 is used in many professional areas such as the new digital television. This compression is an object-oriented compression technique and one of the most efficient. As the adaptive technique has to be used during the compression, one has to build a MPEG-4 codec following the Moving Picture Experts Group (MPEG) requirements. As the work was not to increase the compression mode, one created a basic MPEG-4 codec (using a professionally-built codec should increase the quality of picture at the reception end). As explained previously, multimedia transmission is a huge bandwidth user, and one knows that IEEE 802.15.1 standard has very low power consumption which actually means a short and limited bandwidth available to transmit information. Hence the only way to allow multimedia to be transmitted correctly through a limited bandwidth is to make the data fit perfectly in the bandwidth. This means finding a solution to stock data when there is too much traffic or when a transmission problem occurs and then to "complete" the bandwidth when the amount of information is once more below the bandwidth maximum rate value. By spreading the information, the token bucket regains tokens as fast as possible and then gives more space for more new data in case of peaks or interferences during communication. To allow the data to be treated and stocked, one added a buffer just before the token bucket. Arguably this addition will increase the delay, but in fact the delay added is not noticeable as the system managed to use the "added buffer" at the same time as the token bucket. Thus the delay will be not due to the buffer itself. Adding learning algorithm will of course create some delay on transmission of the data to the sender device. But the delay created will not reduce the quality of the picture at the reception; on the contrary, this delay will increase the quality of service concerning the data loss prevention. In fact the efficiency of the transmission had increased by sending the maximum data possible according the bandwidth rate and the data available. This information can be proved by the noticeably reduced standard deviation which means this system is using more accurately the available bandwidth. This will then reduce the data loss caused by overflowing token bucket and will facilitate retransmission if a problem appears during communication.

The method one decided to follow to highlight the results is to compare two different designs. One uses a standard Variable Bit Rate (VBR) to transmit the data through IEEE 802.15.1 standard, and the second uses a hybrid neural fuzzy algorithm associated to an added buffer to transmit the data through the wireless device. Two movie clips have been used for the simulation. First one transmitted the data through the simulated device, and then one simulated the reception part. The results are shown in chapter 5. In these results one could sometimes see the rate of the data transmitted being over the actual maximum rate of the device. In other words, during this simulation, the amount of data sent to the device happened to be too high compared with the bandwidth available. When this occurs some of the data is stored in the token bucket (which is part of the IEEE 802.15.1 hardware). It is not a problem to put some data in the token bucket until the token bucket is empty (no more tokens inside, means no more space available to data). If too much data arrived for a too long period, an overflow appears at the token bucket and that results in data loss.

After the first set of simulations on the VBR system, one decided to implement the design of adaptive and intelligent techniques. On this design, one decided to implement a new buffer between the output of the MPEG-4 encoder and the token bucket. This buffer, called "added buffer", and the token bucket are managed by two fuzzy logic controllers and a neural network was used to manage the data transmitted in order to avoid the overflows previously demonstrated. One developed and discussed the results shown on the chapter 5. To continue the comparison between the two systems, one used the same principle used for the VBR simulation, with the hybrid neural fuzzy algorithm added. The results shown previously are obvious – the adaptive algorithms reduce significantly the data loss and the excessive delay.

6.2 Conclusion

The project presented, developed a new design to enable video to be transmitted over IEEE 802.15.1 standard. This standard has a short bandwidth (724 Kb/s download and 54 Kb/s upload in the asynchronous mode), which would not be enough to transmit a MPEG-4 compressed file as a video needs a huge bandwidth to be transmitted with enough quality of pictures. In this project, one came up with the idea of adding some artificial intelligence in order to smooth the transmission by spreading the data on the bandwidth to optimise the transmission. A new buffer has been added to the token bucket already in the IEEE 802.15.1 communication device. This "added buffer" helps the token bucket by not being overflowed. The neural network and fuzzy logic controllers analyse the traffic and adapt the amount of data transmitted in order to spread information and reach the mean value of the bandwidth.

These results confirm the idea that applying artificial intelligence on IEEE 802.15.1 standard can improve QoS by developing data loss prevention. By reducing the standard deviation, the system increases the efficiency of the bandwidth; this has for effects to improve the quality of picture by reducing the data loss. The system is using this hybrid intelligent technique to optimise the data sent into the IEEE 802.15.1 device, this is then reducing by more than forty percent the standard deviation during the transmission compare to the standard system. This reduction is very significant and allows more data to be transmitted by using the whole bandwidth available even if the information to transmit is not using it. This improvement prevents data loss which increases a more efficient transmission that increases the quality of picture on the screen of the receiver.

Another point very helpful is this design can be developed on commercial products like web-cameras. Some web-cameras already have wireless communication system but all are based on Wi-Fi, which has an enough bandwidth but a consequent power consumption that is not really suitable for portable devices. Battery power is a big issue for portable devices and IEEE 802.15.1 is very gentle with the battery as it uses very low power. Hence the algorithm implemented on the IEEE 802.15.1 wireless device to transmit MPEG-4 video could provide a better quality of picture than a classic device.

6.3 Contributions

- **Conception of a hybrid neural-fuzzy algorithm:**

one developed a hybrid scheme to transmit MPEG-4 video with a sufficient quality of picture at the receiving end. This design comprises of a fuzzy logic set of rules and an artificial neural network algorithms. This research uses an 'added-buffer' to prevent excessive data loss of MPEG-4 video over IEEE

802.15.1 transmission and subsequently increases picture quality. The hybrid scheme regulates the output rate of the added-buffer to ensure that MPEG-4 video stream conforms to the traffic conditions of the IEEE 802.15.1 channel during the transmission period, that is to send more data when the bandwidth is not fully used and keep the data in the buffers if the bandwidth is overused. Computer simulation results confirm that intelligence techniques and added-buffer do improve quality of picture, reduce data loss and communication delay, as compared with conventional MPEG video transmission over IEEE 802.15.1.

- **Improvement of the quality of picture by amplitude and standard deviation reduction using artificial neural network and rule based fussy logic techniques:**

During this research one proved the amplitude reduction between the hybrid neural-fuzzy system and a conventional system, a reduction of at least thirty percent between the learning system and the usual design has been shown. This information is really important in term of bandwidth efficiency as one is proving here that my design reduces the standard deviation which is going to help to decrease the data loss, which consequently will improve the quality of picture at the receiving end. The prevention of data loss is one of the most important aspects to deal with in order to obtain a better QoS.

• **Increasing quality of service by optimising the stream to conform to the traffic conditions:**

Another part of this research is based on the fact that with the degree of starvation of the buffers, one can optimise the bandwidth used during a transmission. As in a first stage of this research an updated picture of transmission state has been made, one can regulate the flow of information sent by the device following the bandwidth requirement. The bandwidth usage varies constantly so an adaptive method has been deployed to follow as good as possible the real time transmission rate. The artificial neural network here will follow instantly the requirement of the transmission system by using the buffers level and allowing more or less data to go through the device to be sent. An amazing forty percent reduction is shown on this research, confirming the amelioration of the QoS in term of data loss prevention. The hybrid system here optimises the bandwidth usage to prevent data loss then increases the efficiency. This results in a better quality of picture at the receiving end.

6.4 Recommendation for further work

During the period of research, methods of video compression have been moving continuously. This means new and more efficient encoding techniques appear that can decrease the data and maintain a sufficient quality of picture at the receiving end. Also new intelligent techniques have been designed for different areas that can be applied to video transmission over short-range wireless networks. However, in the evolution of multimedia compression techniques, MPEG-4 is still for the moment the most competitive codec and is the standard used for most professional activities (digital TV, video conferencing, etc...). It is actually interesting to keep this encoder in order to be implanted easily on existing devices. Obviously, using a different compression technique, a new design of the whole system would be necessary, which would mean increasing the price and increasing the time taken for the development.

The proposal is to implement some new and possibly more efficient intelligent technique to enable controllers to be more accurate in the analysis of the real-time transmission. one does not explain any new design at this stage. The purpose is to highlight some new research done during recent times and that can be used to enhance our design by increasing the quality of picture at the receiving end. A new adaptive technology has been developed in order to focus on propagating information. Can this neural propagation be used for multimedia wireless transmission over IEEE 802.15.1? Some research could be carried out on this topic, to investigate whether, by using a more specific intelligent technique to spread the information on the bandwidth, it would be possible to control transmission problem and decrease data loss more quickly. As the rapidity of communication and quality of picture is an important problem, this solution could enhance this work by eliminating the burstiness. This new solution also may be more efficient in terms of power consumption, as the data will stay in the buffer for a shorter time due to a more efficient training. Such an idea could be developed in some other research, as this design proves that this technique is useful in this domain.

References

[1] E. Aboelela and C. Douligeris. Routing in multimetric networks using a fuzzy link cost. In Proceedings of the International Symposium on Computers & Communications ISCC'97, pages 397–401, June 1997. IEEE Computer Society.

[2] G. Ascia, V. Catania, G. Ficili, S. Palazzo, and D. Panno. A VLSI fuzzy expert system for real-time traffic control in ATM networks. IEEE Transactions on Fuzzy Systems, volume 5 issue 1, pages 20–31, February 1997.

[3] G. Ascia, V. Catania, G. Ficili, S. Palazzo, and D. Panno. A comparative analysis of fuzzy versus conventional policing mechanisms for ATM networks. IEEE/ACM Transactions on Networking, volume 4 issue 3, pages 449–459, June 1996.

[4] C. Chang and R. Cheng. Traffic control in an ATM network using fuzzy set theory. In Proceedings of the IEEE INFOCOM'94 Conference, pages 1200–1207, Toronto, Canada, June 1994. IEEE Communications Society.

[5] R-G. Cheng and C-J. Chang. Design of a fuzzy traffic controller for ATM networks. IEEE/ACM Transactions on Networking, volume 4 issue 3, pages 460–469, June 1996.

[6] C. Douligeris and G. Develekos. A fuzzy logic approach to congestion control in ATM networks. In ProcIcc95 , pages 1969–1973.

[7] C. Douligeris and G. Develekos. Neuro-fuzzy control in ATM networks. IEEE Communications Magazine, pages 154–162, May 1997.

[8] I. W. Habib. Applications of neurocomputing in traffic management of ATM networks. Proceedings of the IEEE, volume 84 issue 10, pages 1430–1441, October 1996.

[9] Q. Hu, D. W. Petr, and C. Braun. Self-tuning fuzzy traffic rate control for ATM networks. Converging Technologies for Tomorrow's Applications IEEE International Conference, volume 1, pages 424-428, 1996

[10] Y. C. Liu and C. Douligeris. Static vs. adaptive feedback congestion controller for ATM networks. Global Telecommunications Conference, volume: 1, pages 291-295, 1995.

[11] E. Nordstrom, J. Carlstrom, O. Gallmo, and L. Asplund. Neural networks for adaptive traffic control in ATM networks. IEEE Communications Magazine, volume 33 issue 10, pages 43–49, October 1995.

[12] Y.-K. Park and G. Lee. Applications of neural networks in high-speed communication networks. IEEE Communications Magazine, volume 33 issue 10, pages 68–74, October 1995.

[13] A. Pitsillides, Y. A. Sekercioglu and G. Ramamurthy. Effective control of traffic flow in ATM networks using fuzzy logic based explicit rate marking (FERM). IEEE Journal on Selected Areas in Communications, volume 15 issue 2, pages 209–225, February 1997.

[14] F. Olufade, W. Onifade, G. Aderounmu, Sustaining Negotiated QoS in Connection Admission Control for ATM Networks using Fuzzy Logic Techniques, Issues in Informing Science and Information Technology, University of Ibadan, volume 5, 2008.

[15] A. A. Tarraf and I. W. Habib. A novel neural network traffic enforcement mechanism for ATM networks. IEEE Journal on Selected Areas in Communications, volume 12 issue 6, pages 1088–1096, August 1994.

[16] A. A. Tarraf, I.W. Habib, and T. N. Saadawi. Reinforcement learning-based neural network congestion controller. In Proceedings of the Military Communications Conference MILCOM'95, volume 2, pages 668–672. IEEE, 1995.

[17] A. Vasilakos, C. Ricudis, K. Anagnostakis, W. Pedrycz, and A. Pitsillides. Evolutionary-fuzzy prediction for strategic QoS routing in broadband networks. In Proceedings of the IEEE World Congress on Computational Intelligence WCCI'98, pages 1488–1493, 1998.

[18] Deng-Feng Li. An approach to fuzzy multi-attribute decision making under uncertainty, Information Sciences: an International Journal, volume 169 issue 1-2, pages 97–112, 2005.

[19] J. J. Astrain, J. Villadangos, J. R. Garitagoitia, J. R. González de Mendívil, and V. Cholvi. Fuzzy location and tracking on wireless networks, Proceedings of the 4th ACM international workshop on mobility management and wireless access, pages: 84 – 91, October 2006, Association for Computing Machinery.

[20] Chenn-Jung Huang, Wei-Kuang Lai, Hsu-Sheng Tsai, Sheng-Yu Hsiao, and Hao-Yu Liu. A fuzzy-routing-zone-based routing protocol for Bluetooth, MANET, Design and Application of Hybrid Intelligent Systems, January 2003, IOS Press.

[21] Bongkarn Homnan, and Watit Benjapolakul, Adaptation of CDMA Soft Handoff Thresholds using Fuzzy Inference System, Wireless Personal Communications: An International Journal, volume 26 issue 4: pages 325–334, 2003

[22] J.J. Astrain, J. Villadangos, M. Castillo, J.R. Garitagoitia, and F. Farina. Mobility Management in Cellular Communication Systems using Fuzzy Systems, Personal Wireless Communications, LNCS 3260, pages 79–91, 2004.

[23] Khaldoun Al Agha and Houda Labiod, MA-WATM: A new approach towards an adaptive wireless ATM network, Mobile Networks and Applications, volume 4 issue 2, pages 101-109, May 1999.

[24] Samuel Moyle, and Michael Watts, Fuzzy neural networks in a palm environment. 18th National Conference on Artificial Intelligence, American Association for Artificial Intelligence, pages 1008 – 1009, July 2002, American Association for Artificial Intelligence.

[25] Liang Cheng. Wireless awareness for wireless intelligent network, Communications, 2003. APCC 2003. The 9th Asia-Pacific Conference volume 2, pages 561–564, 21–24 September 2003.

[26] Mahdi Nasereddin, Abdullah Konak, and Michael R. Bartolacci, A neural network-based approach for predicting connectivity in wireless networks, International Journal of Mobile Network Design and Innovation, volume 1 issue 1, pages: 18–23, , October 2005.

[27] T. D. Ndousse, Fuzzy neural control of voice cells in ATM networks. IEEE Journal on Selected Areas in Communications, volume 12 issue 9, pages 1488–1494, December 1994.

[28] Rose Q Hu and David W. Petr, A Predictive Selftuning Fuzzy-logic Feedback Rate Controller. IEEE/ACM Transactions on networking, volume 8 issue 6, pages 697-709, 2000.

[29] Martin Heckmann, Frédéric Berthommier, and Kristian Kroschel, Noise adaptive stream weighting in audio-visual speech recognition. EURASIP Journal on Applied Signal Processing, volume 2002 issue 1, pages 1260–1273, January 2002.

[30] Somchai Lekcharoen, Chalida Chaochanchaikul, and Chanintorn Jittawiriyanukoon. A design for fuzzy control policing mechanisms on quality of service support in wireless networks, Mobility '06: Proceedings of the 3rd International Conference on Mobile Technology, Applications & Systems, article no. 48, October 2006.

[31] You-Chang Ko, Sun-Chun Park, Chong-Youn Chun, Hyong-Woo Lee, and Choong-Ho Cho. An Adaptive QoS Provisioning Distributed Call Admission Control using Fuzzy Logic Control, ICC 2001 – IEEE International Conference on Communications, no. 1, pages 356-360, 2001.

[32] Claudio E. Palazzi, Giovanni Pau, Marco Roccetti, Stefano Ferretti, and Mario Gerla. Wireless home entertainment centre: Reducing last hop delays for real-time applications, ACE '06: Proceedings of the 2006 ACM SIGCHI International Conference on Advances in Computer Entertainment Technology, article no. 67, June 2006.

[33] T. Ahmed, A. Nafaa, and A. Mehaoua. An Object-Based MPEG-4 Multimedia Content Classification Model for IP QoS Differentiation, ISCC '03: Proceedings of the Eighth IEEE International Symposium on Computers and Communications, IEEE Computer Society, pages 1091- 1096, June 2003.

[34] Alex Bordetsky, Kevin Brown, and Leann Christianson. Adaptive Management of QoS Adaptive Management of QoS Requirements for Wireless Multimedia Communications, Information Technology and Management, volume 4 issue 1, pages: 9 – 31, January 2003

[35] Mohammad Saraireh, Reza Saatchi, Samir Al-khayatt, and Rebecca Strachan. Assessment and improvement of quality of service in wireless networks using fuzzy and hybrid genetic-fuzzy approaches, Artificial Intelligence Review, volume 27 issue 2-3, pages: 95 – 111, 2007.

[36] Chao-Lieh Chen, and Po-Chien Hsiao. Supporting QoS in wireless MAC by fuzzy control, IEEE Wireless Communications and Networking Conference, volume 2, pages: 1242–1247, 2005,

[37] A. Yener, and C. Rose. Genetic algorithms applied to cellular call admission: Local policies, IEEE Transactions on Vehicular Technology, volume 46 issue 1, pages 72–79, 1997.

[38] Janaki Bandara, Xuemin Shen, and Zafrin Nurmohamed. Resource Allocator for Non Real-Time Traffic in Wireless Networks using Fuzzy Logic, Wireless Personal Communications: an International Journal, volume 21 issue 3, pages 329 – 344, June 2002

[39] Manus Kwan, Kutluyil Doançay, and Lakhmi Jain. Fair multi-path selection for real-time video transmission in ad-hoc networks using artificial intelligence, design and application of hybrid intelligent systems, pages: 830 - 841, 2003.

[40] H. Kazemian. A fuzzy approach to MPEG video transmission in ATM networks. Fuzzy Sets and Systems – An International Journal in Information Science and Engineering, volume 157 issue 16, pages 2259–2272, August 2006.

[41] P. Koutsakis, and M. Paterakis. Call-Admission-Control and traffic-policing mechanisms for the transmission of videoconference traffic from MPEG-4 and H.263 video coders in wireless ATM networks. IEEE Transactions on Vehicular Technology, volume 53 issue 5, pages 1525–1530, 2004.

[42] Sofoklis A. Kyriazakos, George T. Karetsos, Ivan Mura, Evangelos Gkroustiotis, Nikolaos Papaoulakis, Charis Kechagias, and Kostas Vlahodimitropoulos. A Real-Time Monitoring and Intelligent Decisions-Making Platform for Enhanced Radio Resource Management in Wireless Cellular Systems, Wireless Personal Communications: An International Journal, volume 30 issue 2-4, pages 77 – 95, September 2004.

[43] Anthony J. Solon, Kevin Curran, and Paul Mc Kevitt. Bandwidth determined transmoding through fuzzy logic in mobile intelligent multimedia presentation systems, Artificial Intelligence Review, volume 26 issue 4, pages: 325 – 363, December 2006.

[44] Z. Jiang and L. Kleinrock. A packet selection algorithm for adaptive transmission of smoothed video over a wireless channel, Journal of Parallel and Distributed Computing, volume 60 issue 4, pages 494–509, 2000.

[45] J. Y. Khan, J. Wall, and M. A. Rashid. Bluetooth-based Wireless Personal Area Network for Multimedia Communication, DELTA '02: Proceedings of the First IEEE International Workshop on Electronic Design, IEEE Computer Society, pages 47-51 January 2002.

[46] R. Razavi, M. Fleury, and M. Ghanbari. Fuzzy logic control of adaptive ARQ for video distribution over a bluetooth wireless link, Advances in Multimedia, volume 2007 issue 1, pages 8 – 21, January 2007.

[47] H. Kazemian and L. Meng. A fuzzy control scheme for video transmission in Bluetooth wireless. INS - Information Sciences, volume 176 issue 9, pages 1266 – 1289, May 2006.

[48] H. Kazemian H. and L. Meng. An adaptive control for video transmission over Bluetooth. IEEE Transactions on Fuzzy Systems, volume 14 issue 2, pages 263–274, April 2006.

[49] V. Bhaskaran and K. Konstantinides. Image and Video Compression Standards, ISBN:0792399528, 1997.

[50] A. Barjatya. Block matching algorithms for motion estimation, Utah state University DIP 6620, 2004.

[51] http://www.medialab.tfe.umu.se/publications/master_thesis/2006/ mpeg-4_style_object_based_codec_with_matlab.pdf, accessed date September 2006.

[52] http://www.bretl.com/mpeghtml/MPEGindex.htm, accessed date December 2004.

[53] I. Richardson. H.264 and MPEG-4 Video Compression Video Coding for Next-generate on Multimedia, ISBN: 978-0-470-84837-1 2003.

[54] M. Ghanbari. Standard codec – Image compression to advanced video coding, ISBN: 0-85296-710-1, 2003.

[55] N. Topiwala. Wavelet image and video compression, ISBN:0792381823, 1998.

[56] A. Bultheel. Wavelets with applications in signal and image processing, Katholieke Universiteit Leuven, 2003.

[57] F.Keinert. Wavelets and multiwavelets, ISBN10 : 1-58488-304-5, 2004.

[58] http://upload.wikimedia.org/wikipedia/commons/thumb/a/a3/Haar_wavelet.png/800px-Haar_wavelet.png, accessed date April 2005.

[59] http://www.math.ohiou.edu/~shen/bookimages/book_figures36.gif, accessed date April 2005.

[60] http://www.math.ohiou.edu/~shen/bookimages/book_figures7.gif, accessed date April 2005.

[61] http://klapetek.cz/wimages/dog_s.jpg, accessed date May 2005.

[62] C. Schremmer. Multimedia Applications of the Wavelet Transform, Mannheim, 2001.

[63] J.D. Villasenor, B. Belzer, and J. Liao, Wavelet filter evaluation for image compression, IEEE transactions of image processing, volume 4 issue 8, pages 1053 – 1060, USA, 1995.

[64] C. Valens, A Really Friendly Guide to Wavelets, http://pagesperso-orange.fr/polyvalens/clemens/wavelets/wavelets.html 1999.

[65] Z. Ya-Qin. Motion-Compensated Wavelet Transform Coding for Color Video Compression, IEEE Transactions on Circuits and Systems for Video Technology, volume: 2 issue 3, pages 285-296, 1992

[66] http://lsiwww.epfl.ch/LSI2001/teaching/webcourse/ch13/Image99.gif, accessed date February 2005.

[67] http://lsiwww.epfl.ch/LSI2001/teaching/webcourse/ch13/ Image100.gif, accessed date February 2005.

[68] http://lsiwww.epfl.ch/LSI2001/teaching/webcourse/ch13/ Image101.gif, accessed date February 2005.

[69] King N. Ngan, Chi W. Yap and Keng T. Tan, Video Coding for Wireless Communication Systems,. Marcel Dekker, page 124, 2001.

[70] http://www.cmi.ac.in/~sdatta/networks/standards/802.15.1-2002.pdf

[71] Http://bluetooth.com/Bluetooth/Technology/Works/Architecture__ Logical_Link_Control_and_Adaptation_Protocol_L2CAP.htm

[72] J. S. Denker. Neural Networks for Computing, American Institute of Physics, New York, 1987.

[73] W. McCulloch, and W. Pitts, A logical calculus of the ideas immanent in nervous activity. Bulletin of Mathematical Biophysics, pages 115–133, 1943.

[74] Simon Haykin. Neural Networks and Learning Machines (3rd edition), ISBN-13: 9780131471399, 2008.

[75] J. L. McClelland, D. E. Rumelhart, and the PDP research group. Parallel distributed processing: Explorations in the microstructure of cognition, volume 2. ISBN: 0-262-13218-4 , 1986.

[76] http://www.learnartificialneuralnetworks.com/perceptronadaline.html, accessed date September 2005.

[77] B. Pearlmutter. Dynamic recurrent neural networks, CMU-CS-88-191, Carnegie Mellon University, pittsburgh, 1990.

[78] J. J. Hopfield, Neural networks and physical systems with emergent collective computational abilities, National Academy of Science USA, volume 79 issue 8, pages 2554–2558, 1982.

[79] J. Anderson, J. Silverstein. Distinctive features, categorical perception, and probability learning: Some applications of a neural model, ISBN:0-262-01097-6, pages: 283 - 325 , 1988.

[80]. D. O. Hebb, The Organization of Behavior, Wiley, New-York, 1949.

[81] http://en.wikipedia.org/wiki/Lotfi_Zadeh, accessed date January 2008.

[82] http://www.mathworks.com/matlabcentral/fileexchange/907-bluetooth-voice-transmission, accessed date December 2005.

[83] http://www.mathworks.com/products/fuzzylogic/, accessed date May 2006.

[84] http://en.wikipedia.org/wiki/Huffman_coding, accessed date July 2006.